



Центр сертификатов доступа

Aladdin Enterprise Certificate Authority Certified Edition

Руководство администратора. Часть 2. Приложение 3. Описание
методов REST API

Издание	RU.АЛДЕ.03.01.020
Документ	RU.АЛДЕ.03.01.020 32 01-2
Версия	2.2.0
Листов	177
Дата	22.05.2025

Авторские права, товарные знаки, ограничения

Данный документ, включая подбор и расположение иллюстраций и материалов в нём, является субъектом авторских прав и охраняется в соответствии с законодательством Российской Федерации. Обладателем исключительных авторских и имущественных прав является АО «Аладдин Р.Д.».

Использование этих материалов любым способом без письменного разрешения правообладателя запрещено и может повлечь ответственность, предусмотренную законодательством РФ. При перепечатке и использовании данных материалов либо любой их части ссылки на АО «Аладдин Р.Д.» обязательны.

Владельцем зарегистрированных товарных знаков "Аладдин", Aladdin, JaCarta, JMS, JAS, Secret Disk, SecurLogon, "Крипто БД", логотипов и правообладателем исключительных прав на их дизайн и использование, патентов на соответствующие продукты является АО «Аладдин Р.Д.».

Названия прочих технологий, продуктов, компаний, упоминающиеся в данном документе, могут являться товарными знаками своих законных владельцев.

Ограничение ответственности

Информация, приведённая в данном документе, предназначена исключительно для ознакомления и не является исчерпывающей. Состав продуктов, компонент, их функции, характеристики, версии, доступность и пр. могут быть изменены АО «Аладдин Р.Д.» без предварительного уведомления.

АО «Аладдин Р.Д.» не гарантирует ни отсутствия ошибок в данном документе, ни того, что описанное программное обеспечение (ПО) не

содержит дефектов, будет работать в произвольно выбранных условиях и при этом удовлетворять всем требованиям, которые могут быть к нему предъявлены.

АО «Аладдин Р.Д.» не гарантирует работоспособность нелегально полученного программного обеспечения. Нелегальное использование программного обеспечения и документации на него преследуется по закону.

Все указанные данные о характеристиках продуктов основаны на международных или российских стандартах и результатах тестирования, полученных в независимых тестовых или сертификационных лабораториях, либо на принятых в компании методиках. В данном документе АО «Аладдин Р.Д.» не предоставляет никаких ни явных, ни подразумеваемых гарантий.

АО «Аладдин Р.Д.» НЕ НЕСЁТ ОТВЕТСТВЕННОСТИ (КАК В СИЛУ ДОГОВОРА, ГРАЖДАНСКОГО ПРАВОНАРУШЕНИЯ, ВКЛЮЧАЯ ХАЛЯТНОСТЬ, ТАК И В ЛЮБОЙ ИНОЙ ФОРМЕ) ПЕРЕД ВАМИ ИЛИ ЛЮБОЙ ТРЕТЬЕЙ СТОРОНОЙ ЗА ЛЮБЫЕ ПОТЕРИ ИЛИ УБЫТКИ (ВКЛЮЧАЯ КОСВЕННЫЕ, ФАКТИЧЕСКИЕ ИЛИ ПОБОЧНЫЕ УБЫТКИ), ВКЛЮЧАЯ БЕЗ ОГРАНИЧЕНИЙ ЛЮБЫЕ ПОТЕРИ ИЛИ УБЫТКИ ПРИБЫЛЬНОСТИ БИЗНЕСА, ПОТЕРЮ ДОХОДНОСТИ ИЛИ РЕПУТАЦИИ, УТРАЧЕННУЮ ИЛИ ИСКАЖЁННУЮ ИНФОРМАЦИЮ ИЛИ ДОКУМЕНТАЦИЮ ВСЛЕДСТВИЕ ИСПОЛЬЗОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И/ИЛИ ЛЮБОГО КОМПОНЕНТА ОПИСАННОГО ПРОДУКТА, ДАЖЕ ЕСЛИ АО «Аладдин Р.Д.» БЫЛО ПИСЬМЕННО УВЕДОМЛЕНО О ВОЗМОЖНОСТИ ПОДОБНЫХ УБЫТКОВ.

Лицензионное соглашение

Пожалуйста, внимательно прочитайте данное лицензионное соглашение прежде, чем использовать содержимое данного комплекта и/или прежде, чем загружать или устанавливать программное обеспечение.

Все указания по использованию программного обеспечения, предоставляемые Закрытым акционерным обществом "Аладдин Р.Д." (или любым его дочерним предприятием – каждое из них упоминаемое как "компания"), подчиняются и будут подчиняться условиям, оговоренным в данном соглашении. Загружая данное программное обеспечение (как определено далее по тексту) и/или устанавливая данное программное обеспечение на Ваш компьютер и/или используя данное программное обеспечение иным способом, Вы принимаете данное соглашение и соглашаетесь с его условиями.

Если Вы не согласны с данным соглашением, не загружайте и/или не устанавливайте данное программное обеспечение и незамедлительно (не позднее 7 (семи) дней с даты ознакомления с настоящим текстом) верните этот продукт в АО «Аладдин Р.Д.», удалите данное программное обеспечение и все его части со своего компьютера и не используйте его никоим образом.

Настоящее лицензионное соглашение (далее "Соглашение") является договором, заключённым между Вами (физическим или юридическим лицом) — конечным пользователем (далее "Пользователь") — и АО «Аладдин Р.Д.» (далее "Компания") относительно передачи неисключительного права на использование настоящего программного обеспечения, являющегося интеллектуальной собственностью Компании.

Права и собственность

ДАННОЕ СОГЛАШЕНИЕ НЕ ЯВЛЯЕТСЯ СОГЛАШЕНИЕМ О ПРОДАЖЕ. Программное обеспечение, включая все переработки, исправления, модификации, дополнения, обновления и/или усовершенствования к нему (далее по всему тексту и любой его части определяемое как Программное обеспечение или ПО), и связанная с ним документация предназначена НЕ ДЛЯ ПРОДАЖИ и является и остаётся исключительной собственностью Компании.

Все права на интеллектуальную собственность (включая, без ограничений, авторские права, коммерческую тайну, товарные знаки, и т.д.), подтверждённые или включённые в приложенные/взаимосвязанные/имеющие отношение к данному руководству, данные, содержащиеся в нём, а также все права на ПО являются и будут являться собственностью исключительно Компании.

Данное соглашение не передаёт Вам права на Программное обеспечение, а лишь предоставляет ограниченное право на использование, которое подлежит отмене согласно условиям данного Соглашения. Ничто в данном Соглашении не подтверждает отказ Компании от прав на интеллектуальную собственность по какому бы то ни было законодательству.

Лицензия

Компания настоящим предоставляет Вам, а Вы получаете индивидуальное, неисключительное и отменяемое ограниченное право на использование данного ПО только в форме исполняемого кода, как

описано в прилагаемой к ПО технической/эксплуатационной документации, и только в соответствии с условиями данного Соглашения:

Вы можете установить ПО и использовать его на компьютерах, расположенных в пределах Вашего предприятия, как описано в соответствующей технической/эксплуатационной документации ПО и в настоящем соглашении.

Вы можете добавить/присоединить Программное обеспечение к программам для мобильных устройств с единственной целью, описанной в данном Соглашении. Принимая условия настоящего соглашения, Вы соглашаетесь:

- не использовать, не модифицировать и не выдавать сублицензии на данное Программное обеспечение и любое другое ПО Компании, за исключением явных разрешений в данном Соглашении;
- не модифицировать, не демонтировать, не декомпилировать, не реконструировать, не видоизменять и не расширять данное Программное обеспечение и не пытаться раскрыть (получить) исходные коды данного Программного обеспечения;
- не помещать данное Программное обеспечение на сервер с возможностью доступа к нему третьих лиц через открытую сеть;
- не использовать какие бы то ни было резервные или архивные копии данного Программного обеспечения (или позволять кому-либо ещё использовать такие копии) с любой иной целью, кроме замены его оригинального экземпляра в случае его разрушения или наличия дефектов.

Требования к использованию

Программное обеспечение должно использоваться и обслуживаться строго в соответствии с описаниями и инструкциями Компании, приведёнными в данном и других документах Компании, в том числе на портале онлайн документации для разработчиков Компании (<http://developer.aladdin-rd.ru/>).

Использование ПО

Пользователь вправе:

- воспроизводить ПО путём записи его в память электронно-вычислительных машин Пользователя, ограниченное правом установки, копирования и запуска программ для ЭВМ;
- встраивать ПО любым способом в продукты и решения Пользователя;
- распространять ПО любым способом исключительно в составе продуктов и решений Пользователя.

При использовании и распространении ПО Пользователь обязан руководствоваться действующим законодательством Российской Федерации и международным законодательством, учитывая ограничения и дополнительные требования, которые могут возникнуть в связи с экспортом шифровальных (криптографических) средств с территории Российской Федерации и импортом таких средств в другие страны. В частности, ограничения и дополнительные требования могут возникать при распространении ПО через магазины приложений, содержащие различные приложения для мобильных устройств.

Обеспечения не будет превышать суммы, выплаченной вами АО «Аладдин Р.Д.» за это ПО.

Условия использования, изложенные в настоящем соглашении, действуют в отношении всего содержимого ПО, в частности в отношении: - дизайна (графики, расположения элементов оформления и т.п.);

- всех иных элементов, в том числе изображений, фонограмм, текстов.

Получаемые Пользователем неисключительные имущественные права не включают права на передачу третьим лицам каких-либо прав на встраивание, воспроизведение, распространение и использование программ для ЭВМ не в составе продуктов и решений Пользователя.

Компания сохраняет за собой все исключительные права на ПО и входящие в него компоненты, включая права на предоставление неисключительных и исключительных прав третьим лицам.

Пользователь вправе осуществлять использование ПО в пределах, предусмотренных настоящим Соглашением, исключительно на территории Российской Федерации.

Обслуживание и поддержка

Компания не несёт обязательств по предоставлению поддержки, обслуживания, модификации или выходу новых релизов ПО.

Ограниченная гарантия

Компания гарантирует, что программное обеспечение с момента приобретения его Вами в течение 12 (двенадцати) месяцев будет функционировать в полном соответствии с его технической/эксплуатационной документацией, при условии, что ПО будет использоваться на компьютерном аппаратном обеспечении и с операционной системой, для которой оно было разработано.

Отказ от гарантии

Компания не гарантирует, что программное обеспечение будет соответствовать Вашим желаниям и требованиям, или что его работа будет бесперебойной или безошибочной. В объёме, предусмотренном законодательством РФ, компания открыто отказывается от всех гарантий, не оговоренных здесь, от всех иных подразумеваемых гарантий. Ни один из дилеров, дистрибьюторов, продавцов, агентов или сотрудников компании не уполномочен производить модификации, расширения или дополнения к данной гарантии.

Если Вы произвели какие-либо модификации ПО или любой из его частей во время гарантийного периода, ПО подверглось повреждению, неосторожному или неправильному обращению, если Вы нарушили любое из условий настоящего Соглашения, то гарантия, упомянутая выше в разделе 5, будет немедленно прекращена.

Гарантия недействительна, если ПО используется в сочетании с иным аппаратным и/или программным обеспечением, отличным от описанных в технической/эксплуатационной документации, или используется на компьютере с любым установленным нелицензионным программным обеспечением.

Ограничение возмещения

В случае нарушения гарантии, оговоренной выше, Компания может по собственному усмотрению:

- заменить ПО, если это не противоречит вышеупомянутому ограничению гарантии;

- возместить стоимость, выплаченную Вами за ПО.

Гарантийные требования должны быть выставлены в письменном виде в течение гарантийного периода, но не позднее 7 (семи) дней с момента обнаружения дефекта, и содержать в себе подтверждения, удовлетворяющие Компанию. Всё ПО (все экземпляры, имеющиеся у Вас) должно быть возвращено Компании и отправлено возвращающей стороной с оплаченной стоимостью перевозки и, при необходимости, страховки. Экземпляры ПО должны быть отправлены с копией платёжных документов и накладных. Исключение косвенных убытков

Стороны признают, что Программное обеспечение не может быть полностью лишено ошибок. Компания не несёт ответственности (как в силу договора, гражданского правонарушения, включая халатность, так и в любой иной форме) перед Вами или любой третьей стороной за любые потери или убытки (включая косвенные, фактические, побочные или потенциальные убытки), включая, без ограничений, любые потери или убытки прибыльности бизнеса, потерю доходности или репутации, утраченную или искажённую информацию или документацию вследствие какого-либо использования данного программного обеспечения и/или любой компоненты данного ПО, даже если компания письменно уведомлена о возможности подобных убытков.

Ограничение ответственности

В случае если, несмотря на условия данного соглашения, компания признана ответственной за убытки на основании каких-либо дефектов или несоответствия программного обеспечения Вашим ожиданиям, полная ответственность за каждый экземпляр дефектного программного

Прекращение действия соглашения

В случае невыполнения Вами условий данного Соглашения действие Вашей лицензии и настоящего Соглашения будет прекращено.

После прекращения действия данного Лицензионного соглашения:

- лицензия, предоставленная Вам данным Соглашением, прекращает своё действие, и Вы после её прекращения не сможете продолжать дальнейшее использование данного Программного обеспечения и других лицензионных Продуктов;

- вы немедленно вернёте в Компанию все экземпляры ПО и все копии такого и/или сотрёте/удалите любую информацию, содержащуюся в электронном виде.

Применимое законодательство

Данное Соглашение должно быть истолковано и определено в соответствии с законодательством Российской Федерации (за исключением конфликта применения правовых норм), и только российский суд уполномочен осуществлять правосудие в любых конфликтах и спорах, вытекающих из данного Соглашения. Невозможность для любой из сторон воспользоваться любым из прав, предоставленных ей по данному Соглашению, или принять меры против другой стороны в случае любого нарушения своих обязательств по Соглашению не должно рассматриваться как отказ этой стороны от последующего понуждения к признанию своих прав или совершению последующих действий в случае дальнейших нарушений.

Государственное регулирование и экспортный контроль

Вы соглашаетесь с тем, что ПО не будет Вами поставляться, передаваться или экспортироваться в какую-либо страну, а также использоваться каким-либо противоречащим закону и условиям настоящего соглашения образом. ПО является предметом дополнительного экспортного контроля, относящегося к Вам или Вашей юрисдикции. Вы гарантируете, что будете соблюдать накладываемые ограничения на экспорт и реэкспорт ПО.

Разное

Настоящее Соглашение представляет собой полное соглашение, относящееся к данной лицензии, и может быть изменено только посредством письменного соглашения, подписанного обеими сторонами. Если выполнение какого-либо условия настоящего Соглашения представляется невозможным, такое условие будет скорректировано только в пределах, обеспечивающих возможность выполнения данного условия.

Я ПРОЧИТАЛ И ПОНЯЛ НАСТОЯЩЕЕ ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ, И СОГЛАСЕН ВЫПОЛНЯТЬ ВСЕ ЕГО УСЛОВИЯ. Я ПРИНИМАЮ ДАННОЕ ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ ЦЕЛИКОМ. ЕСЛИ Я НЕ ПРИНИМАЮ ЭТО ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ ИЛИ ХОТЯ БЫ ОДИН ИЗ ЕГО ПУНКТОВ, ТО ДАННОЕ ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ НЕ ВСТУПАЕТ В СИЛУ, И Я ОБЯЗУЮСЬ НЕ УСТАНОВЛИВАТЬ И НЕ ИСПОЛЬЗОВАТЬ ДАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.

Содержание

Методы идентификации и аутентификации	8
1.1 Метод идентификации и аутентификации по сертификату доступа.....	8
1.1.1 Пример использования метода для аутентификации по сертификату доступа:.....	8
1.1.2 Пример ответа на запрос к данному методу:.....	8
1.2 Метод обновления маркера доступа	9
1.2.1 Пример использования метода	9
1.2.2 Пример ответа на запрос к данному методу:.....	9
1.3 Метод обновления последней активности учетной записи	9
Методы работы с субъектами.....	11
1.4 Метод поиска субъектов.....	11
1.4.1 Пример использования метода	12
1.4.2 Пример ответа на обращение к методу	12
1.5 Метод получения субъекта по идентификатору.....	15
1.5.1 Пример использования метода	16
1.5.2 Пример ответа на обращение к методу	16
1.6 Метод создания и изменения субъекта	17
1.7 Методы создания и изменения субъекта на основании запроса pkcs#10.....	18
1.7.1 Метод создания и изменения субъекта на основании запроса pkcs#10 (формат запроса на сертификат из входных параметров – multipart/form-data).....	18
1.7.2 Метод создания и изменения субъекта на основании запроса pkcs#10 (формат запроса на сертификат из входных параметров – application/json).....	20
1.8 Метод удаления субъекта.....	22
1.8.1 Пример использования метода	22
1.9 Метод поиска ID субъектов	22
Методы работы с шаблонами сертификатов	24
1.10 Метод поиска шаблонов	24
1.10.1 Пример использования метода (указан текст «web» для поиска):	25
1.10.2 Пример ответа на обращение к методу	25
1.11 Метод получения шаблона по идентификатору	26
1.11.1 Пример использования метода.....	27
1.11.2 Пример ответа на обращение к методу.....	28
Методы работы с Центрами сертификации.....	40
1.12 Метод получения активного ЦС.....	40
1.12.1 Пример использования метода.....	42
1.12.2 Пример ответа на обращение к методу.....	42
1.13 Метод получения ЦС по идентификатору.....	44
1.13.1 Пример использования метода.....	46
1.13.2 Пример ответа на обращение к методу.....	46
1.14 Метод получения Центров сертификации.....	49
1.14.1 Пример использования метода.....	51
1.14.2 Пример ответа на обращение к методу.....	51
Методы работы с сертификатами.....	56
1.15 Метод выпуска сертификата в контейнере pkcs#12.....	56
1.15.1 Пример использования метода.....	57

1.15.2 Пример ответа на обращение к методу.....	58
1.16 Методы выпуска сертификата по запросу pkcs#10.....	59
1.16.1 Выпуск сертификата по запросу pkcs#10 (формат запроса на сертификат из входных параметров – multipart/form-data).....	59
1.16.2 Выпуск сертификата по запросу pkcs#10 (формат запроса на сертификат из входных параметров – application/json).....	61
1.17 Методы перевыпуска сертификата по запросу pkcs#10.....	63
1.17.1 Перевыпуск сертификата по запросу pkcs#10 (формат запроса на сертификат из входных параметров – multipart/form-data).....	63
1.17.2 Перевыпуск сертификата по запросу pkcs#10 (формат запроса на сертификат из входных параметров – application/json).....	65
1.18 Методы валидации запроса pkcs#10.....	67
1.18.1 Метод валидации запроса pkcs#10 (формат запроса на сертификат из входных параметров – multipart/form-data).....	67
1.18.2 Метод валидации запроса pkcs#10 (формат запроса на сертификат из входных параметров – application/json).....	69
1.19 Метод поиска сертификатов.....	70
1.19.1 Пример использования метода.....	72
1.19.2 Пример ответа на обращение к методу.....	72
1.20 Метод получения сертификата по идентификатору.....	74
1.20.1 Пример использования метода.....	76
1.20.2 Пример ответа на обращение к методу.....	76
1.21 Метод получения сертификата по серийному номеру.....	79
1.21.1 Пример использования метода.....	81
1.21.2 Пример ответа на обращение к методу.....	82
1.22 Метод получения сертификата по его отпечатку.....	85
1.22.1 Пример использования метода.....	87
1.22.2 Пример ответа на обращение к методу.....	88
1.23 Метод отзыва (приостановки) сертификата по идентификатору.....	91
1.24 Метод активации сертификата по идентификатору.....	92
1.25 Метод публикации сертификата в РС по идентификатору.....	92
1.26 Метод расшифровки контейнера сертификата.....	92
1.26.1 Пример использования метода.....	94
1.26.2 Пример ответа на обращение к методу.....	96
1.27 Метод расшифровки сертификата.....	98
1.27.1 Пример использования метода.....	101
1.27.2 Пример ответа на обращение к методу.....	101
1.28 Метод расшифровки запроса на сертификат.....	104
1.28.1 Пример использования метода.....	105
1.28.2 Пример ответа на обращение к методу.....	106
Методы экспорта файлов.....	107
1.29 Метод получения сертификата по идентификатору сертификата.....	107
1.29.1 Пример использования метода.....	107
1.29.2 Пример ответа на обращение к методу.....	107
1.30 Метод получения запроса на сертификат по идентификатору сертификата.....	108
1.31 Метод получения цепочки сертификата по идентификатору сертификата.....	108

1.32 Метод получения контейнера PKCS#12 по идентификатору сертификата	108
1.33 Метод получения сертификата Центра сертификации по идентификатору Центра сертификации.....	109
1.34 Метод получения цепочки сертификатов Центра сертификации по идентификатору Цentra сертификации	109
1.35 Метод получения CRL по идентификатору Центра сертификации	109
1.36 Метод получения DeltaCRL по идентификатору Центра сертификации	109
Методы работы с точками распространения	111
1.37 Метод генерации и публикации CRL по идентификатору Центра сертификации	111
1.38 Метод генерации и публикации CRL по идентификатору Центра сертификации (устаревший)	111
Методы работы с точками подключения и ресурсными системами	112
1.39 Метод поиска зарегистрированных ресурсных систем	112
1.40 Метод получения ресурсной системы по идентификатору	112
1.41 Метод полной синхронизации ресурсной системы	113
1.42 Метод поиска точек подключения	113
1.43 Метод получения точки подключения по идентификатору	113
1.44 Метод частичной синхронизации точки подключения.....	114
Методы получения версии сервисов	115
1.45 Метод получения версии сервиса обратной совместимости	115
1.45.1 Пример использования метода.....	115
1.45.2 Пример ответа при успешном использовании данного метода представлен ниже:	115
1.46 Метод получения версии сервиса внешних интеграций	115
Методы работы с Syslog-серверами	117
1.47 Метод поиска Syslog-серверов.....	117
1.48 Метод получения Syslog-сервера по идентификатору.....	117
1.49 Метод создания Syslog-сервера.....	117
1.50 Метод обновления Syslog-сервера.....	118
1.51 Метод деактивации Syslog-сервера	118
1.52 Метод активации Syslog-сервера	119
1.53 Метод удаления Syslog-сервера	119
Методы работы с учетными записями	120
1.54 Метод поиска учетных записей	120
1.55 Метод получения учетной записи по идентификатору	120
1.56 Метод получения учетной записи по отпечатку сертификата.....	121
Методы работы с издателями	122
1.57 Метод поиска издателей.....	122
Методы работы с подразделениями и группами безопасности	123
1.58 Метод поиска подразделений.....	123
1.59 Метод получения подразделения по идентификатору.....	123
1.60 Метод групп безопасности.....	124
1.61 Метод получения группы безопасности по идентификатору	124
Дополнительные возможности.....	126
1.62 Выполнение администратором методов от имени другой учетной записи.....	126
Сопоставление методов обратной совместимости и публичного REST API.....	127
Диаграмма последовательности получения сертификата по запросу PKCSN°10.....	128

1.63 Диаграмма последовательности получения сертификата по запросу PKCS#10128

Спецификация OpenAPI132

Обозначения и сокращения176

лист регистрации изменений177

МЕТОДЫ ИДЕНТИФИКАЦИИ И АУТЕНТИФИКАЦИИ

1.1 Метод идентификации и аутентификации по сертификату доступа

POST API – Аутенти фикация с помощью сертификата	
URL – x509-provider-service/api/v2a/public/auth/sign-in/x509	
Swagger: отсутствует, так как метод реализован на уровне прокси (отсутствует в реализации сервиса)	
Query	
-	
Request	
-	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
token (string),	Маркер доступа
refresh (string)	Токен обновления
}	

1.1.1 Пример использования метода для аутентификации по сертификату доступа:

```
sudo curl --request POST -k --cert cert.crt.pem --cert-type PEM --key test.key.pem --key-type PEM 'https://localhost/x509-provider-service/api/v2a/public/auth/sign-in/x509'
```

- В примере указаны: `cert.crt.pem` – сертификат открытого ключа; `test.key.pem` – закрытый ключ.
- Ключ и сертификат можно получить из контейнера `p12`, используя утилиту `openssl`:

```
openssl pkcs12 -in INITIAL_ADMIN.p12 -out test.key.pem -nocerts -nodes
openssl pkcs12 -in INITIAL_ADMIN.p12 -out cert.crt.pem -clcerts -nokeys
```

1.1.2 Пример ответа на запрос к данному методу:

```
{
  "status": 200,
  "data": {
    "token": "eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiJiYzZmMDQ5ZC1hM2EwLTQ4ZjQtYjE0Ny02YmYzNmRjYjZhNGIiLCJpYXQiOiJlMDE5ODU2MjYsImV4cCI6MTcwMTk4NTgwNn0uYX7pbvPHtP8tQ0vg9upBR7AlYXbgwr8XX1tldggcySzehA6qi2mJGE-MlMf0iNeAOy0L_QJ7YVsB_080yTwcw2E6ipsTRI1rRbq1VHKw4xTNcVh2UVCXXTxz4WaYw3uSkNJfPPLgs7wyxyAlu3Is4d7JuiCSgONoj-rArjam_ccTuEHJnHp7PTnrDLOUNm8elu-VkP9qvQPqLb0P_K7-XLp3lQrWOZWMPPM4imldia70fBD4QeNiruQjYHK4ENbqCARIMiPnV0GFTz7JMnLLP1liFrnOgHhUUhp6JylKOTKxddLgUjTU03BmucUwAl_ccThZZPzpwNLd6Cx67X0Og",
    "refresh": "8b73de98-721d-4d0b-98f7-f19745272f0c"
  }
}
```

В результате успешной аутентификации сервер возвращает cookies-файл с маркером доступа и токеном обновления. Для дальнейшей аутентификации используется данный маркер доступа (см. примеры использования методов далее).

Маркер доступа необходимо передать в http-header "token", либо приложить к запросу полученный cookies-файл с маркером доступа (http-header "cookie").

1.2 Метод обновления маркера доступа

POST API – Обновление токена доступа	
Метод доступен администратору и оператору	
URL – security-service/api/v2a/public/auth/refresh-token	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a]Контроллер%3A Авторизации/refreshToken	
Query	
-	
Request	
-	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
token (string),	Маркер доступа
refresh (string)	Токен обновления
}	

1.2.1 Пример использования метода

```
curl --location -k --request POST 'https://172.17.152.213/security-service/api/v2a/ui/auth/refresh-token' --header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI2YTZhNmM4Yi0zYWQyLTQwYjMtYWI5NS05NDgxNmZlYjBkZGQiLCJpYXQiOiJlZ3M0IiwiaWF0IjE5MTQ1MTQ0LjUucxBCuiClMwfmVNoYy8peliYgdDw4eQ-v4Blp5mYRcOxSOswGipUGZ_0cTrV3QyVXXrty9CHTM2virdnvU7xp_4lASug5MRUCjOhsWnNFIGu21bDoFbK7jQPDDdG5LfCxfswQVAXrY-G4BPst8azPdY-CVtsjXIPyUmkL-vo7_7b4Prs2OwNeewYn_9KsdHXV56XJHEo4nBM0wsew830Q813pFe7LpLXNuQ8SvekEr9QKGSspBgYT3qC9D_57NhWzEgKqVRsLxk4f4p8x7TI4zIsFI1v1VaqrnAiDnElwxFHNnvXVDuyxh54g6BkLfoHj3DGnCiijiopPcti_pw; refresh=7aaad966-b417-418e-b2bb-63574ddcc8d1'
```

1.2.2 Пример ответа на запрос к данному методу:

```
{"status":200,"data":{"token":"eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI2YTZhNmM4Yi0zYWQyLTQwYjMtYWI5NS05NDgxNmZlYjBkZGQiLCJpYXQiOiJlZ3M0IiwiaWF0IjE5MTQ1MTQ0LjUucxBCuiClMwfmVNoYy8peliYgdDw4eQ-v4Blp5mYRcOxSOswGipUGZ_0cTrV3QyVXXrty9CHTM2virdnvU7xp_4lASug5MRUCjOhsWnNFIGu21bDoFbK7jQPDDdG5LfCxfswQVAXrY-G4BPst8azPdY-CVtsjXIPyUmkL-vo7_7b4Prs2OwNeewYn_9KsdHXV56XJHEo4nBM0wsew830Q813pFe7LpLXNuQ8SvekEr9QKGSspBgYT3qC9D_57NhWzEgKqVRsLxk4f4p8x7TI4zIsFI1v1VaqrnAiDnElwxFHNnvXVDuyxh54g6BkLfoHj3DGnCiijiopPcti_pw; refresh=7aaad966-b417-418e-b2bb-63574ddcc8d1","refresh":"3071f779-f4f7-47db-91a4-13acfc50005"}}
```

1.3 Метод обновления последней активности учетной записи

PUT API – Обновление последней активности учетной записи по ее идентификатору	
Метод должен быть доступен только администратору	
URL – security-service/api/v2a/public/accounts/{accountId}/activity	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/%5Bv2a%5D%20Контроллер%3A%20Учетные%20записи/updateActivity	
Query	
{(id – обязательный параметр)}	

id (UUID)	Идентификатор учетной записи
}	
Request	
-	
Response	
-	

МЕТОДЫ РАБОТЫ С СУБЪЕКТАМИ

1.4 Метод поиска субъектов

GET API – Поиск субъектов	
Метод доступен администратору и оператору	
URL – subjects-service/api/v2a/public/subjects	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a	
Контроллер%3A субъекты/findAll_2	
Query	
{	
search (string) [опционально],	Полнотекстовый поиск (имя субъекта)
isBlocked (boolean) [опционально],	Флаг: субъект заблокирован в ресурсной системе
isConnected (boolean) [опционально],	Флаг: субъект подключен к ресурсной системе
id (UUID[]) [опционально],	ID субъекта
notId (UUID[]) [опционально],	Исключая ID субъекта
securityGroupId (UUID[]) [опционально],	ID группы безопасности
organizationalUnitId (UUID[]) [опционально],	ID структурной единицы
resourceId (UUID[]) [опционально],	ID ресурсной системы
sortDirection (string) [опционально],	Направления сортировки (ASC;DESC)
sortBy (string[]) [опционально],	Список полей, к которым применяется сортировка
pageOffset (integer) [опционально],	Смещение от начала списка (пагинация)
pageLimit (integer) [опционально]	Ограничение на размер выборки (пагинация)
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID субъекта
commonName (string)	Имя субъекта
distinguishedName (string),	Расположение субъекта в ресурсной системе
resource: {	Ресурсная система
id (uuid),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string)	BaseDN точки подключения к ресурсной системе
},	
subjectName: {	Имя субъекта
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): {	Поля разделенного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
subjectAltName: {	Альтернативное имя субъекта
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): {	Поля альтернативного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
isConnected (boolean),	Флаг: субъект подключен к ресурсной системе

isBlocked (boolean),	Флаг: субъект заблокирован в ресурсной системе
certificatesCount (integer),	Количество сертификатов
modify (instant),	Время изменения (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
sid (string)	SID субъекта или NULL
}	

1.4.1 Пример использования метода

```
curl -k --request GET --location 'https://172.17.152.213/subjects-
service/api/v2a/public/subjects?pageLimit=2147483647&pageOffset=0&sortDirection=asc'
\
--header
'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI4ZTcxMzc2MS1hMGE4LTQ1MjQtYTE5Z
i1lOTkxMjNiMDk0YTMiLCJpYXQiOiJlMjIwNDY4MjQsImV4cCI6MTcwMjA0NzAwNH0.1m7blLyaiztUrZ2LIh
vRFpbW2160iw_ELeiEzDRcjRXV9H_D_WLZiyk1Ft4Z3G8pDd4EAXODgd6eVRCsO8yPwEKLOZwSl1IFjxRRza
-3g9ohrgXtLddkdWn-LSRcO_LVrLT-
vUiQrvRXpbC0rBoWcoXMgCFnYErIngvFZmZxreP1PvQU9f9hLbszp1xm6Aso6239VG8VfF9b04Qtwhg_7kPmY
1NvE8pRhflr7ICTjcMOMKHC-bdJ1yxKmK_-xtFzdirjEu0WAEXwKTxY6sdMBidcyqRTiKBldX0klzfZm-
_veRI4XoWhHTZrQxmp86me2vy_JLn-AsMWj3D7DdAmg'
```

1.4.2 Пример ответа на обращение к методу

```
{
  "status": 200,
  "data": {
    "range": {
      "count": 2,
      "offset": 0,
      "limit": 2147483647,
      "totalPages": 1,
      "pageNumber": 0
    },
    "items": [
      {
        "id": "000d2c40-cb30-4e9b-9429-934302389328",
        "commonName": "Lewis Hance",
        "distinguishedName": "CN=Lewis Hance,OU=created-
users,DC=winad,DC=local",
        "resource": {
          "id": "4c651fcc-94ba-4dea-9dc5-d9d2b2334cc9",
          "commonName": "winad",
          "distinguishedName": "DC=winad,DC=local"
        },
        "subjectName": {
          "EMAILADDRESS": {
            "values": [
              "Carating@test.local",
              "LewisKHance@einrot.com"
            ],
            "editable": false
          },
          "CN": {
            "values": [
              "Lewis Hance"
            ],

```

```

        "editable": false
    },
    "SURNAME": {
        "values": [
            "Hance"
        ],
        "editable": false
    },
    "GIVENNAME": {
        "values": [
            "Lewis"
        ],
        "editable": false
    },
    "NAME": {
        "values": [
            "Lewis Hance"
        ],
        "editable": false
    }
},
"subjectAltName": {
    "RFC822NAME": {
        "values": [
            "Carating@test.local",
            "LewisKHance@einrot.com"
        ],
        "editable": false
    },
    "MS_UPN": {
        "values": [
            "Carating@test.local"
        ],
        "editable": false
    }
},
"isConnected": true,
"isBlocked": false,
"modify": "2023-06-16T00:59:24Z",
"updated": "2024-02-05T21:00:07.617169Z",
"created": "2024-02-01T05:08:09.379555Z"
},
{
    "id": "0011005f-bb95-4162-ad6e-015ae8cac7be",
    "commonName": "Megan Mejorado",
    "distinguishedName": "CN=Megan Mejorado,OU=created-
users,DC=winad,DC=local",
    "resource": {
        "id": "4c651fcc-94ba-4dea-9dc5-d9d2b2334cc9",
        "commonName": "winad",
        "distinguishedName": "DC=winad,DC=local"
    },
    "subjectName": {
        "EMAILADDRESS": {
            "values": [

```

```
        "Yourbithes45@test.local",
        "MeganMMejorado@dayrep.com"
    ],
    "editable": false
},
"CN": {
    "values": [
        "Megan Mejorado"
    ],
    "editable": false
},
"SURNAME": {
    "values": [
        "Mejorado"
    ],
    "editable": false
},
"GIVENNAME": {
    "values": [
        "Megan"
    ],
    "editable": false
},
"NAME": {
    "values": [
        "Megan Mejorado"
    ],
    "editable": false
}
},
"subjectAltName": {
    "RFC822NAME": {
        "values": [
            "Yourbithes45@test.local",
            "MeganMMejorado@dayrep.com"
        ],
        "editable": false
    },
    "MS_UPN": {
        "values": [
            "Yourbithes45@test.local"
        ],
        "editable": false
    }
},
"isConnected": true,
"isBlocked": false,
"modify": "2023-06-16T03:45:57Z",
"updated": "2024-02-05T21:00:22.612777Z",
"created": "2024-02-01T05:08:38.131896Z",
"sid": "NULL"
}
]
}
```

В примере выше (вызов осуществлялся с маркером доступа пользователя с ролью «Администратор») получены все имеющиеся субъекты программы.

1.5 Метод получения субъекта по идентификатору

GET API – Получение субъекта по идентификатору	
Метод доступен администратору и оператору при наличии полномочий на управление субъектом, идентификатор которого передается во входных параметрах	
URL – subjects-service/api/v2a/public/subjects/{id}	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A субъекты/findById_2	
Query	
{	
id (UUID)	ID субъекта
}	
Request	
-	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID субъекта
commonName (string)	Имя субъекта
distinguishedName (string),	Расположение субъекта в ресурсной системе
resource: {	Ресурсная система
id (uuid),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string)	BaseDN точки подключения к ресурсной системе
},	
subjectName: {	Имя субъекта
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): {	Поля разделенного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
subjectAltName: {	Альтернативное имя субъекта
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): {	Поля альтернативного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
isConnected (boolean),	Флаг: субъект подключен к ресурсной системе
isBlocked (boolean),	Флаг: субъект заблокирован в ресурсной системе
certificatesCount (integer),	Количество сертификатов
modify (instant),	Время изменения (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
sid (string)	SID субъекта или NULL
}	

1.5.1 Пример использования метода

```
curl -k --request GET --location 'https://172.17.152.213/subjects-  
service/api/v2a/public/subjects/0002db60-2364-43cc-a7d9-14390aaf2628' \  
--header  
'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiJkYzgzMDI2OC04MDA5LTQ2Y2EtOWNlY  
i03MzEyMWNiOGFhYzYiLCJpYXQiOiJlZ3MDIyNzY4NzIsImV4cCI6MTcwMjI3NzA1Mn0.TBJk1CmffoE8u2chrh  
3-4lJhzlg5YA36NjSn0i30liJRLJkVYPxQ6QmQ8iBoq08jIiPi3SM_C7d0pi7yaljbylcobWmLzSTDnGeZ-  
aYP44qI4VbZvPtKFPeJb9GebJoOE5rGgl9RAPMGT74YVNJzlcZ_GlNUP2HP3BPLGzm3-  
3BSH3yImXlvaSoZRsZLYmIKreHv6oD_UhwmcAigFW2WYkcOxNgDvHWyOHar_xekQuK3jVsUERkvF19rNdssJd  
W5n0eThMnGlBZhXp_OWKnfyi6V9NFialDocW_HtrJfwlbnNbAnKuEImbylE6aDjPfS5XRjVuNZstIAgl4sLVC9  
yg'
```

1.5.2 Пример ответа на обращение к методу

```
{
  "status": 200,
  "data": {
    "id": "0002db60-2364-43cc-a7d9-14390aaf2628",
    "commonName": "Mary Washington",
    "distinguishedName": "CN=Mary Washington,OU=created-users,DC=winad,DC=local",
    "resource": {
      "id": "4c651fcc-94ba-4dea-9dc5-d9d2b2334cc9",
      "commonName": "winad",
      "distinguishedName": "DC=winad,DC=local"
    },
    "subjectName": {
      "EMAILADDRESS": {
        "values": [
          "Saighterse84@test.local",
          "MaryRWashington@einrot.com"
        ],
        "editable": false
      },
      "CN": {
        "values": [
          "Mary Washington"
        ],
        "editable": false
      },
      "SURNAME": {
        "values": [
          "Washington"
        ],
        "editable": false
      },
      "GIVENNAME": {
        "values": [
          "Mary"
        ],
        "editable": false
      },
      "NAME": {
        "values": [
          "Mary Washington"
        ],

```

```

        "editable": false
    },
    "subjectAltName": {
        "RFC822NAME": {
            "values": [
                "Saighterse84@test.local",
                "MaryRWashington@einrot.com"
            ],
            "editable": false
        },
        "MS_UPN": {
            "values": [
                "Saighterse84@test.local"
            ],
            "editable": false
        }
    },
    "isConnected": true,
    "isBlocked": false,
    "modify": "2024-02-05T10:09:47Z",
    "updated": "2024-02-05T21:00:27.282374Z",
    "created": "2024-02-01T05:08:36.038636Z",
    "sid": "NULL"
}
}

```

1.6 Метод создания и изменения субъекта

PUT API – Создание и изменение субъекта	
Метод доступен администратору и оператору при наличии полномочий	
URL – subjects-service/api/v2a/public/subjects	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a	
Контроллер%3A субъекты/update	
Query	
-	
Request	
{	
id (UUID) [опционально],	Идентификатор субъекта
subjectName: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] } [опционально],	Поля разделенного имени субъекта
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] }	Поля альтернативного имени субъекта

} [опционально]	
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID субъекта
commonName (string)	Имя субъекта
distinguishedName (string),	Расположение субъекта в ресурсной системе
resource: {	Ресурсная система
id (uuid),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string)	BasedN точки подключения к ресурсной системе
},	
subjectName: {	Имя субъекта
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): {	Поля разделенного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
subjectAltName: {	Альтернативное имя субъекта
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): {	Поля альтернативного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
isConnected (boolean),	Флаг: субъект подключен к ресурсной системе
isBlocked (boolean),	Флаг: субъект заблокирован в ресурсной системе
certificatesCount (integer),	Количество сертификатов
modify (instant),	Время изменения (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
sid (string)	SID субъекта или NULL
}	

1.7 Методы создания и изменения субъекта на основании запроса pkcs#10

1.7.1 Метод создания и изменения субъекта на основании запроса pkcs#10 (формат запроса на сертификат из входных параметров – multipart/form-data)

PUT API – Создание и изменение субъекта на основании запроса pkcs#10 (multipart/form-data)	
Метод доступен администратору и оператору при наличии полномочий	
URL – subjects-service/api/v2a/public/subjects/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A субъекты/updateByPkcs10AsMultipartFile_1	
Query	
-	
Request	
{	

Id (UUID) [опционально],	Идентификатор субъекта
request (binary),	Файл запроса на сертификат (см. пример использования метода ниже)
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] } [опционально]	Поля альтернативного имени субъекта
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID субъекта
commonName (string)	Имя субъекта
distinguishedName (string),	Расположение субъекта в ресурсной системе
resource: {	Ресурсная система
id (uuid),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string)	BaseDN точки подключения к ресурсной системе
},	
subjectName: {	Имя субъекта
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): {	Поля разделенного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
subjectAltName: {	Альтернативное имя субъекта
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): {	Поля альтернативного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
isConnected (boolean),	Флаг: субъект подключен к ресурсной системе
isBlocked (boolean),	Флаг: субъект заблокирован в ресурсной системе
certificatesCount (integer),	Количество сертификатов
modify (instant),	Время изменения (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
sid (string)	SID субъекта или NULL
}	

1.7.1.1 Пример использования метода через Curl:

```
curl -k --request PUT --location 'https://192.168.111.18/subjects-  
service/api/v2a/public/subjects/pkcs10' \  
--header 'Cookie:  
token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiaOiI0ZmFhYjgxNi0wZGZhLTQ0ZjctOGQzZS0xNDkwNTdmYzYwZmEiLCJpYXQiOiJlMDEwMjc5ODYsImV4cCI6MTcwMjAyODE2Nn0.EBQw9zAD-I8BsCkYT5A8mUdyjBc7f14Ogm17ui0IdjpdK6Eq4k5UQ13Vgj2kMhU2XQKnBxt4NZNJ45Y8d-
```

```
_FXXwk_1KcG94X8YvrmF6tWHnAOt3DLjsIMm3ZbMJjAopQx-0PQURg8_aQ6d4E2rLWB-
mWE1h8JqcwZ5g29yD1qHS7V0ArSy_AbkFmnqUrrq6CVcQLETGef05qCIIX5_WEB9bF-
bnevA9aqLTPWdYzbcXTAUgRumVkeKgi_if9S-
mSYFC8EtYvztC30klEtssR6wG2VucIozLhYRuWVavXD7OXyQ8pRtKxYIQimt7cuD5Ez7v-
ML9LfDZ5B6EIh1OM7g' \
--form 'request=@"/C:/Users/user/Downloads/request.csr"'
```

1.7.2 Метод создания и изменения субъекта на основании запроса pkcs#10 (формат запроса на сертификат из входных параметров – application/json)

Примечание: отличается от метода выше форматом передаваемого во входных параметрах запроса на сертификат. В данном методе на вход поступает содержимое файла запроса на сертификат в формате PEM (содержимое в Base64).

PUT API – Создание и изменение субъекта на основании запроса pkcs#10 (application/json)	
Метод доступен администратору и оператору при наличии полномочий	
URL – subjects-service/api/v2a/public/subjects/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A субъекты/updateByPkcs10AsMultipartFile_1	
Query	
-	
Request	
{	
Id (UUID) [опционально],	Идентификатор субъекта
request: {	Файл запроса на сертификат
contentType (string) [опционально],	Тип загружаемого файла (HTTP MediaType) – application/octet-stream)
fileName (string) [опционально],	Имя загружаемого файла
data (string:binary)	Содержимое PEM файла запроса на сертификат (массив байт в Base64) – см. пример использования метода ниже
},	
subjectAltName: {	Поля альтернативного имени субъекта
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD): string[]	
} [опционально]	
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID субъекта
commonName (string),	Имя субъекта
distinguishedName (string),	Расположение субъекта в ресурсной системе
resource: {	Ресурсы
id (uuid),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string)	BaseDN точки подключения к ресурсной системе
},	
subjectName: {	Имя субъекта
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T,	Поля разделенного имени субъекта

DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): {	
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
subjectAltName: {	Альтернативное имя субъекта
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT IDENTIFICATION METHOD, UNKNOWN): {	Поля альтернативного имени субъекта
values (string[]),	Значения компонента
editable (boolean),	Флаг: компонент доступен для редактирования
}	
},	
isConnected (boolean),	Флаг: субъект подключен к ресурсной системе
isBlocked (boolean),	Флаг: субъект заблокирован в ресурсной системе
certificatesCount (integer),	Количество сертификатов
modify (instant),	Время изменения (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
sid (string)	SID субъекта или NULL
}	

1.7.2.1 Пример использования метода через Curl:

```
curl -k --request PUT --location 'https://172.17.152.213/subjects-  
service/api/v2a/public/subjects/pkcs10' --header 'Content-Type: application/json' --  
data '{  
    "request": {  
        "contentType": "application/octet-stream",  
        "fileName": "test",  
        "data":  
"LS0tLS1CRUdJTiBORVcgQ0VSVElGSUNBVEUgUkVRVUVTVc0tLS0tCk1JSUNnekNDQVdzQ0FRQXdQakVQTUEw  
R0ExVUVCaE1HVW5WemMybGhNUTR3REFZRFZRUUtFd1YwYUhKbFpURU0NCk1Bb0dBmVVFQ3hNRGRIZHZNUTB3Q  
3dZRFZRURFd1JTVDAA5VU1JSUJJakFOQmdrcWhraUc5dzBCQVFFRkFBT0MNCkFROEFNSUlCQ2dLQ0FRRUFpOV  
RJeXRScklFYi9kUlRhSWtxcEdZSGNktlRpB0VNK2E2L090SVk3d3JCjY2M2ceYNcnRlSudGc0VkRmpNcWpKbFB  
lWmFhQ2lEMDhVMU1td0h2VXBqK3AzSEd3YWdIRkk2elNrcnNEUTJnKzhDLlB0ZnENCnVqcU1vRUhNTWZJK3FE  
TEdVemp6eGJUUV0f1T05DVjBtUmNpdTlIU2o0Ynh4dTUVUeW05QnNhYVBvczR6MzAycC8NCk13b3dleWxkZ0EzV  
2NYS1YzVTRsbW1VcERmcmlnNDRiWEw5R3BTejNJVKd1MlkxQWtlc3RpQXNCEjhWMUNTbU8NCiglgrT3hwUUVIMV  
lFVHVsRFpOdGtTM2kvVmthV0JtTTZXK1VJWDZRYUwxTTdRSDFMTWVvYlludlUwSEVmQjRwYysNCnlJN1dKK2t  
Cc3JLK1RNbHEyZFV5RTdnWFR4YUZks1lpSmZKTnVRSURBUUFCb0FBd0RRWUpLb1pJaHZjTkFRRUYNckJRQUrN  
Z0VCQUIiwWHNGYzhYUGJmME9hVGdCVkk2K0R1Res5NZVBVHREZHRCRTEwbDYzd1cwZkhzZmM0WHJ5bloNCkpaT  
XRmaXRGUExxcGtaT1lfBETySGpiZlJzYzd2NzJSRmszbGpUQ1l0Ym9INmFkT1IvWEVYMhlVS1pPM1lTNSsNCn  
lNbE5neERNZ0NzeEhZQlFoNzhOYk1NUWIrSDhxMzh5ZzF1L0lsL3JmOVhQbENCL05Wdm5raS8wOHQvQ0dySVc  
NCjZqb0R0bnhQcFdyYwdQujV5c3I0Q1J6L0lDMkU0cEFESjZQdv1kZW9EYlFUTE1yMGFTRG5NTGdnZGN4eFVL  
NUkNCjNFb01zTEclldUpoZWVQODBSFFFuYnVLbStjTU4rK3lPcUxHa0FaaHdrRGREano0cTAyelV1RnhpeEJBV  
zcrRzYncmkvejZhEmRtTm5MR1VPSGZpYm5qN3Q2aFVKZUsxa289Ci0tLS0tRU5EIE5FVyBDRVJUSUZJQ0FURS  
BSRVFVRVNULS0tLS0K"  
  
    }  
}' --cookie  
'token=eyJhbGciOiJIUzUxMiJ9.eyJzZXNzaW9uSWQiOiJlZjI0ODYyMiljZjRlLTRmZmUtYTJkZiImODdmM  
mViOGRkZTYiLCJpYXQiOiJE3MDiWmJkxMzQsImV4cCI6MTcwMjAyOTMxNH0.ZRrUsOTfw6jJzi4YS8c2xq3Qz2  
EOBNpv2IV8I991m2hJrz6VmTQtzdyKihG7BDUZp_zOf_j63UbRmhtue59o6Wqlii7sBRaFakckzxCP98mTc5m  
uB9E77blfu0OHcoZqZfSX36bSoH6R0hLZ1fyTzjRnsfdrlKisQAaf0XrjK5NsfmgnEQOCinAu2f82tbftxfyS  
1Aegjr1SL6A7EZQ_UvJNffg4FWWygzruSyJlVKAyn_kwdolJelj_pOc4yr_rXH391IqX6GL7tlC5cUSaDCXwg  
-hE49gkw9f9-2oBlip92Puq6dbeAPvPF5nnhH3JHW-3JoqsrUsimTrugfl9GA'
```

1.8 Метод удаления субъекта

DELETE API – Удаление субъекта	
Метод доступен администратору	
URL – subjects-service/api/v2a/public/subjects/{id}	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a] Контроллер%3A субъекты/deleteById	
Query	
{ (id – обязательный параметр)	
id (UUID)	Идентификатор субъекта
}	
Request	
-	
Response	
-	

1.8.1 Пример использования метода

```
curl -k --location --request DELETE 'https://172.17.152.213/subjects-
service/api/v2a/public/subjects/1a254868-5318-11ee-aa77-360c16967470' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiJiNmRlZWUyYS00N2MxLTQzZj
QtYmEyNS05MTk1YWVlMjQzMtKiLCJpYXQiOiJlZ3MDIyNzcnNjQsImV4cCI6MTcwMjI3NzI0NH0.e8b0fI9N91D
QzveJx9Wg3mepQl8trwLS-6ldl8J9-IUKBxRXAIYmidz7EQf-
HAMjjHP5tCrSb71HvLYN6x5_abpIb6oAVLIbw2XLytwlTjQ9dQS_HcrhDjOeelc77fJlWIt-
FlsyMlmsuMYMZFa4d9x71yNsDTNlRTl3bebCojlp4LCJqGFf80qv70-r9jERsSGSSoSGHGMTafm0eG-
cL3zjyR-
Dne8PbyMKAPg8NA043uxLzUTrG4nOrsH7rxmM4yrfR6fueTnpPrkLbVnZb5zM0DU0mXc4YrgDUhsiOhGu_tj8
tazCfAmm0FCe6Tijqh-Tmi52ejpMm3w1TwpJIw'
```

1.9 Метод поиска ID субъектов

GET API – Поиск ID субъектов	
Метод доступен администратору и оператору	
URL – subjects-service/api/v2a/public/subjects/ids	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a] Контроллер%3A субъекты/findAllAsIds	
Query	
{	
search (string) [опционально],	Полнотекстовый поиск (имя субъекта)
isBlocked (boolean) [опционально],	Флаг: субъект заблокирован в ресурсной системе
isConnected (boolean) [опционально],	Флаг: субъект подключен к ресурсной системе
id (UUID[]) [опционально],	ID субъекта
notId (UUID[]) [опционально],	Исключая ID субъекта
securityGroupId (UUID[]) [опционально],	ID группы безопасности
organizationalUnitId (UUID[]) [опционально],	ID структурной единицы
resourceId (UUID[]) [опционально],	ID ресурсной системы
sortDirection (string) [опционально],	Направления сортировки (ASC;DESC)
sortBy (string[]) [опционально],	Список полей, к которым применяется сортировка
pageOffset (integer) [опционально],	Смещение от начала списка (пагинация)
pageLimit (integer) [опционально]	Ограничение на размер выборки (пагинация)
}	

Request	
-	
Response	
ResponseEntity -> CollectionResponse -	Ответ JSON в HTTP-body
> {	
items (UUID)	ID субъекта
}	

МЕТОДЫ РАБОТЫ С ШАБЛОНАМИ СЕРТИФИКАТОВ

Сопоставление идентификаторов шаблонов	
Идентификатор 1.2.0	Идентификатор 2.x
100001	9129245a-eaad-4ebc-a2a4-8845ac0336fb
100002	af3b0355-1798-4c64-98f7-a9c70407db1c
100003	bf2dac0a-f05f-49dd-95b4-e50691489b6a
100004	aa03e458-50cd-46b8-82cd-d5612ed3b647
100005	aac2e49b-9c8e-4869-80c1-eef526ba75ab
100006	059a38f5-f345-4275-b79f-e7e6cc3cbb68
100007	08c66f99-218a-46ef-bdee-6a2b3b26a4f1
100008	0c234243-18cf-4c05-b699-537731b2436f
100009	11ec34a4-d03e-4059-92f0-9c09b08bffeaa
100010	18d9bd4e-6f15-423f-8137-ac8416ad6874

1.10 Метод поиска шаблонов

GET API – Поиск шаблонов	
Метод доступен администратору и оператору. В ответе для оператора должны содержаться только те шаблоны, на использование которых ему предоставлены полномочия.	
URL – templates-service/api/v2a/public/templates	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a]Контроллер%3A шаблоны/findAll	
Query	
{	
types (enum[]: EMBEDDED, CLONED, IMPORTED, UNKNOWN) [опционально],	Тип шаблона
certificateType (enum[]: CERTIFICATE, ROOT_CA, SUB_CA, UNKNOWN) [опционально],	Тип выпускаемого сертификата
endEntityType (enum[]: ROOT_CA, SUB_CA, USER, DEVICE, UNKNOWN) [опционально],	Тип субъекта
search (string) [опционально],	Полнотекстовый поиск по имени шаблона
removed (boolean) [опционально],	Флаг: шаблон удален
id (UUID[]) [опционально],	ID шаблона
notId (UUID[]) [опционально],	Исключая ID шаблона
keyAlgorithm (enum[]: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN) [опционально],	Фильтр: алгоритм ключа включен в шаблоне ¹
extendedKeyUsage (string[]) [опционально],	Фильтр: расширенное использование ключа
sortDirection (string) [опционально],	Направления сортировки (ASC; DESC)
sortBy (string[]) [опционально],	Список полей, к которым применяется сортировка
pageOffset (integer) [опционально],	Смещение от начала списка (пагинация)
pageLimit (integer) [опционально]	Ограничение на размер выборки (пагинация)
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID шаблона
name (string),	Имя шаблона

¹ В случае использования множественных значений для фильтра «keyAlgorithm» в ответе метода будут содержаться шаблоны, в которых включен хотя бы один алгоритм из перечня, указанного в данном фильтре.

type (enum: EMBEDDED, CLONED, IMPORTED, UNKNOWN),	Тип шаблона
certificateAuthorityId (UUID),	ID ЦС, который должен использоваться при выпуске сертификата по данному шаблону
endEntityType (enum[:ROOT_CA, SUB_CA, USER, DEVICE, UNKNOWN]),	Тип типа субъекта
certificateCount (int64),	Число выпущенных по шаблону сертификатов
removed (boolean),	Флаг: шаблон удален
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
appendSubjectSid (boolean)	Флаг: включать SID субъекта в сертификат
}	

1.10.1 Пример использования метода (указан текст «web» для поиска):

```
curl -k --location 'https://172.17.152.213/templates-
service/api/v2a/public/templates?search=web' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiaOiJmOTVjMjUyNy01YjIjLTRkOT
ctYTZhMy0yNTk1NGUzNDZkMDciLCJpYXQiOiJlMjIyNzcyNDgsImV4cCI6MTcwMjI3NzQyOH0.aBGJVubnt9W
iiDbzj7zAIEH3gbGS_qoqzzvRUk2C8xASy86iA7AFw1583wVVjWWVz_nBO5vc-
izdcGfSjmkJmsCGalwFxqt4qQiPE03FTFHAvgHGJ63sqd8uCaUe_hBLxZDu-
EKtiIBKcNvHoBflku42xFuwi9A56bnxsgoc4G0dl0OIM9M9kApbkrKzEyBNoWpmxKLJC6OuZWRsfdCI2n80AH
SnAbwqlgmVZSEIQ-
_26DqLxt9QXSyOfTCC_kDihRA2BodLZYToUitZwHyUxgTOQze9jmSyEDZ7ow2friotTynWSmAnwauif8KTLz
Cxp-Xx742H7pFzqSdUYASjvQ'
```

1.10.2 Пример ответа на обращение к методу

```
{
  "status": 200,
  "data": {
    "range": {
      "count": 2,
      "offset": 0,
      "limit": 2147483647,
      "totalPages": 1,
      "pageNumber": 0
    },
    "items": [
      {
        "id": "059a38f5-f345-4275-b79f-e7e6cc3cbb68",
        "name": "WEB-Client",
        "endEntityType": "USER",
        "type": "EMBEDDED",
        "certificateCount": 1,
        "removed": false,
        "created": "2023-12-08T10:25:34.970782Z",
        "updated": null
      },
      {
        "id": "08c66f99-218a-46ef-bdee-6a2b3b26a4f1",
        "name": "WEB-Server",
        "endEntityType": "DEVICE",
        "type": "EMBEDDED",
        "certificateCount": 1,
        "removed": false,
```

```

        "created": "2023-12-08T10:25:34.970782Z",
        "updated": null
    }
  ]
}
}

```

1.11 Метод получения шаблона по идентификатору

GET API – Получение шаблона по идентификатору	
Метод доступен:	
– администратору; – оператору при наличии полномочий на использование шаблона, идентификатор которого передается во входных параметрах.	
URL – templates-service/api/v2a/public/templates/{id}	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a	
Контроллер%3A шаблоны/findById	
Query	
{	
id (UUID)	ID шаблона
}	
Request	
-	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID шаблона
name (string),	Имя шаблона
type (enum: EMBEDDED, CLONED, IMPORTED, UNKNOWN),	Тип шаблона
certificateType (enum: CERTIFICATE, ROOT_CA, SUB_CA, UNKNOWN),	Тип выпускаемого сертификата
certificateAuthorityId (UUID),	ID ЦС, который должен использоваться при выпуске сертификата по данному шаблону
endEntityType (enum[]:ROOT_CA, SUB_CA, USER, DEVICE, UNKNOWN),	Тип субъекта
removed (boolean),	Флаг: шаблон удален
validity (int64),	Время действия выпускаемого сертификата (мс)
rsa: {	Описание RSA-криптографии
use (boolean),	Флаг: RSA-ключи доступны для шаблона
minLength (int32),	Минимальная длина RSA-ключа
lengths (int32[])	Доступные длины RSA-ключа
},	
ecdsa: {	Описание ESDCA-криптографии
use (boolean),	Флаг: ESDCA -ключи доступны для шаблона
minLength (int32),	Минимальная длина ESDCA -ключа
lengths (int32[])	Доступные длины ESDCA -ключа
},	
gost: {	Описание ГОСТ-криптографии
use (boolean),	Флаг: ГОСТ -ключи доступны для шаблона
minLength (int32),	Минимальная длина ГОСТ -ключа
lengths (int32[])	Доступные длины ГОСТ -ключа
},	
keyUsages: {	Назначение ключа сертификата
critical (boolean),	Флаг: расширение критическое
values (enum[]:DIGITAL_SIGNATURE, NON_REPUDIATION, KEY_ENCIPHERMENT, DATA_ENCIPHERMENT, KEY_AGREEMENT, KEY_CERT_SIGN, CRL_SIGN, ENCIPHER_ONLY, DECIPHER_ONLY, UNKNOWN)	Значение расширения

<code>},</code>	
<code>extendedKeyUsages: {</code>	Расширенное назначение ключа сертификата
<code>critical (boolean),</code>	Флаг: расширение критическое
<code>values (string[])</code>	Значение расширения (OIDs)
<code>},</code>	
<code>policies: {</code>	Политики сертификата
<code>critical (boolean),</code>	Флаг: расширение критическое
<code>values (string[])</code>	Значение расширения (OIDs)
<code>},</code>	
<code>subjectDN: [{</code>	Имя субъекта сертификата
<code>index (int32),</code>	Индекс (для сортировки, по умолчанию – 0)
<code>name (string),</code>	Имя компонента
<code>description (string),</code>	Описание компонента
<code>required (boolean),</code>	Флаг: обязателен к заполнению
<code>validation (boolean),</code>	Флаг: валидация значения
<code>modifiable (boolean),</code>	Флаг: доступен к редактированию
<code>defaultValue (string),</code>	Значение по умолчанию
<code>regex (string),</code>	Регулярное значение для валидации значения
<code>alert (string),</code>	Предупреждение о неудачной валидации значения
<code>code (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN)</code>	Код компонента
<code>}],</code>	
<code>subjectAltName: [{</code>	Расширенное имя субъекта сертификата
<code>index (int32),</code>	Индекс (для сортировки, по умолчанию – 0)
<code>name (string),</code>	Имя компонента
<code>description (string),</code>	Описание компонента
<code>required (boolean),</code>	Флаг: обязателен к заполнению
<code>validation (boolean),</code>	Флаг: валидация значения
<code>modifiable (boolean),</code>	Флаг: доступен к редактированию
<code>defaultValue (string),</code>	Значение по умолчанию
<code>regex (string),</code>	Регулярное значение для валидации значения
<code>alert (string),</code>	Предупреждение о неудачной валидации значения
<code>code (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN),</code>	Код компонента
<code>generalName (int32),</code>	Идентификатор компонента в RFC
<code>oid (string)</code>	OID компонента в RFC
<code>}],</code>	
<code>updated (instant),</code>	Время обновления (ISO 8601)
<code>created (instant)</code>	Время создания (ISO 8601)
<code>appendSubjectSid (boolean)</code>	Флаг: включать SID субъекта в сертификат
<code>}</code>	

1.11.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/templates-
service/api/v2a/public/templates/08c66f99-218a-46ef-bdee-6a2b3b26a4f1' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI3YzE0MmJlZi1jMTNhLTQwOG
UyYWY3Yi1kMDYyNjFkYzQ3ZDMLCJpYXQiOiJlZ3MDIyNzclNzQsImV4cCI6MTcwMjI3NzclNH0.Leza2kpVqv
VPdlqGx2FeETAoeUSfd1dL01W9HSrl0zXd1rukWYtXRlPvo3ce2TDJffHYd9sookfPpSuWYUJL44Dy8t-
B_59Kl8sxGOhtzos4ZXK4LYiXCqpiZlmeiZTXpGwbCPk5ps3S7qSA_ckAAo0hf0ooUrukVDg1ss2Zhld-
i2swQldq5uy5lbRkzSeikq_PYfhnsekTsq6d7fD_7FzBFC1oZUHm8fBM8eXbqA5wK0A3TaW9IPTKQb8AXSnno
```

```
6oTatnG7aJdi6X-KP8Y1vO1XeYMTv1ngvpPhkVdlVC6P9UdD6oSJQuk1z2NChhZZEtI16f-  
4iFXKOFvHN5J0w'
```

1.11.2 Пример ответа на обращение к методу

```
{  
  "status": 200,  
  "data": {  
    "id": "9129245a-eaad-4ebc-a2a4-8845ac0336fb",  
    "name": "Root CA",  
    "validity": 757468800000,  
    "endEntityType": "ROOT_CA",  
    "type": "EMBEDDED",  
    "removed": false,  
    "rsa": {  
      "use": true,  
      "minLength": 1024,  
      "lengths": [  
        1024,  
        1536,  
        2048,  
        3072,  
        4096,  
        6144,  
        8192  
      ]  
    },  
    "ecdsa": {  
      "use": true,  
      "minLength": 256,  
      "lengths": [  
        256,  
        384,  
        521  
      ]  
    },  
    "gost": {  
      "use": true,  
      "minLength": 256,  
      "lengths": [  
        256,  
        512  
      ]  
    },  
    "keyUsages": {  
      "critical": true,  
      "values": [  
        "DIGITAL_SIGNATURE",  
        "KEY_CERT_SIGN",  
        "CRL_SIGN"  
      ]  
    },  
    "extendedKeyUsages": {  
      "critical": false,  
      "values": [  

```

```
        "2.5.29.37.0",
        "1.3.6.1.5.5.7.3.2",
        "1.3.6.1.5.5.7.3.1"
    ]
},
"policies": {
    "critical": false,
    "values": []
},
"subjectDN": [
    {
        "index": 0,
        "name": "Country",
        "description": "Страна",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-z]{2}$",
        "defaultValue": null,
        "alert": null,
        "code": "C"
    },
    {
        "index": 0,
        "name": "Unique Identifier (UID)",
        "description": "Уникальный идентификатор",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "UID"
    },
    {
        "index": 0,
        "name": "Given name",
        "description": "Отчество",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "GIVENNAME"
    },
    {
        "index": 0,
        "name": "Initials",
        "description": "Инициалы",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
```

```
        "alert": null,
        "code": "INITIALS"
    },
    {
        "index": 0,
        "name": "Surname",
        "description": "Фамилия",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "SURNAME"
    },
    {
        "index": 0,
        "name": "Organizational unit",
        "description": "Подразделение",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "OU"
    },
    {
        "index": 0,
        "name": "Locality",
        "description": "Населенный пункт",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "L"
    },
    {
        "index": 0,
        "name": "State or province",
        "description": "Регион",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "ST"
    },
    {
        "index": 0,
        "name": "Domain component",
        "description": "Компонент домена",
```

```
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "DC"
    },
    {
        "index": 0,
        "name": "Postal code",
        "description": "Почтовый индекс",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "POSTALCODE"
    },
    {
        "index": 0,
        "name": "Business category",
        "description": "Категория организации",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "BUSINESSCATEGORY"
    },
    {
        "index": 0,
        "name": "Telephone number",
        "description": "Телефонный номер",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "TELEPHONENUMBER"
    },
    {
        "index": 0,
        "name": "Pseudonym",
        "description": "Псевдоним",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "PSEUDONYM"
    }
```

```
    },
    {
      "index": 0,
      "name": "Postal address",
      "description": "Почтовый адрес",
      "required": false,
      "validation": true,
      "modifiable": true,
      "regex": null,
      "defaultValue": null,
      "alert": null,
      "code": "POSTALADDRESS"
    },
    {
      "index": 0,
      "name": "Street",
      "description": "Адрес",
      "required": false,
      "validation": true,
      "modifiable": true,
      "regex": null,
      "defaultValue": null,
      "alert": null,
      "code": "STREET"
    },
    {
      "index": 0,
      "name": "Name",
      "description": "Имя",
      "required": false,
      "validation": true,
      "modifiable": true,
      "regex": null,
      "defaultValue": null,
      "alert": null,
      "code": "NAME"
    },
    {
      "index": 0,
      "name": "Title",
      "description": "Заголовок",
      "required": false,
      "validation": true,
      "modifiable": true,
      "regex": null,
      "defaultValue": null,
      "alert": null,
      "code": "T"
    },
    {
      "index": 0,
      "name": "Description",
      "description": "Описание",
      "required": false,
      "validation": true,
```

```
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "DESCRIPTION"
    },
    {
        "index": 0,
        "name": "Common name",
        "description": "Общее имя",
        "required": true,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "CN"
    },
    {
        "index": 0,
        "name": "Organization",
        "description": "Организация",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "O"
    },
    {
        "index": 0,
        "name": "ИНН",
        "description": "Идентификационный номер налогоплательщика",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^\\d{12}$",
        "defaultValue": null,
        "alert": null,
        "code": "INN"
    },
    {
        "index": 0,
        "name": "ОГРН",
        "description": "Основной государственный регистрационный номер",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^\\d{13}$",
        "defaultValue": null,
        "alert": null,
        "code": "OGRN"
    },
    {
        
```

```
        "index": 0,
        "name": "ОГРНИП",
        "description": "Основной государственный регистрационный номер
индивидуального предпринимателя",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^\\d{15}$",
        "defaultValue": null,
        "alert": null,
        "code": "OGRNIP"
    },
    {
        "index": 0,
        "name": "СНИЛС",
        "description": "Страховой номер индивидуального лицевого счёта",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^\\d{11}$",
        "defaultValue": null,
        "alert": null,
        "code": "SNILS"
    },
    {
        "index": 0,
        "name": "ИНН ЮЛ",
        "description": "ИНН юридического лица",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^\\d{10}$",
        "defaultValue": null,
        "alert": null,
        "code": "INNLE"
    },
    {
        "index": 0,
        "name": "Unstructured address",
        "description": "Неструктурированный адрес",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "UNSTRUCTUREDADDRESS"
    },
    {
        "index": 0,
        "name": "Unstructured name",
        "description": "Неструктурированное имя",
        "required": false,
        "validation": true,
        "modifiable": true,
```

```
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "UNSTRUCTUREDNAME"
    },
    {
        "index": 0,
        "name": "Serial number",
        "description": "Серийный номер",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-z0-9'()+,\\-\\.\\/:\\\\\\\\=? ]+$",
        "defaultValue": null,
        "alert": null,
        "code": "SERIALNUMBER"
    },
    {
        "index": 0,
        "name": "Domain qualifier",
        "description": "Квалификатор доменного имени",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-z0-9'()+,\\-\\.\\/:\\\\\\\\=? ]+$",
        "defaultValue": null,
        "alert": null,
        "code": "DN"
    },
    {
        "index": 0,
        "name": "Email Address (E)",
        "description": "Электронный адрес",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-zA-Яа-я0-9._-]+@[A-Za-zA-Яа-я0-9._-]+$",
        "defaultValue": null,
        "alert": null,
        "code": "EMAILADDRESS"
    }
],
"subjectAltName": [
    {
        "index": 0,
        "name": "Uniform resource identifier",
        "description": "Идентификатор универсального ресурса",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "UNIFORM_RESOURCE_ID",
        "generalName": 6,
```

```

        "oid": null
    },
    {
        "index": 0,
        "name": "Service Name",
        "description": "Имя сервиса",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": null,
        "defaultValue": null,
        "alert": null,
        "code": "SRV_NAME",
        "generalName": 0,
        "oid": "1.3.6.1.5.5.7.8.7"
    },
    {
        "index": 0,
        "name": "RFC 822 Name",
        "description": "RFC 822 Name",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-zA-Яа-я0-9._-]+@[A-Za-zA-Яа-я0-9._-]+$",
        "defaultValue": null,
        "alert": null,
        "code": "RFC822NAME",
        "generalName": 1,
        "oid": null
    },
    {
        "index": 0,
        "name": "DNS Name",
        "description": "Доменное имя",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-zA-Яа-я0-9-.*]+$",
        "defaultValue": null,
        "alert": null,
        "code": "DNS_NAME",
        "generalName": 2,
        "oid": null
    },
    {
        "index": 0,
        "name": "IP address",
        "description": "IP адрес",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^( [0-9A-Fa-f]{1,4}: ){7} [0-9A-Fa-f]{1,4}$|^ (\\d{1,3}\\.){3} \\d{1,3}$",
        "defaultValue": null,
        "alert": null,

```

```

        "code": "IPADDRESS",
        "generalName": 7,
        "oid": null
    },
    {
        "index": 0,
        "name": "Registered Identifier (OID)",
        "description": "Зарегистрированный идентификатор",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^(([0-1]\\.[1-9])|([2]\\.[1-9]\\d*)|([0-1]\\.[1-3]\\d))((\\.0)|\\.([1-9]\\d*))*$",
        "defaultValue": null,
        "alert": null,
        "code": "REGISTERED_ID",
        "generalName": 8,
        "oid": null
    },
    {
        "index": 0,
        "name": "Directory Name",
        "description": "Наименование директории",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex":
        "^(?: (?:EMAILADDRESS|CN|UID|SERIALNUMBER|OU|O|L|ST|C|T|SURNAME|STREET|INITIALS|GIVENNAME|DC|UNSTRUCTUREDADDRESS|UNSTRUCTUREDNAME|POSTALCODE|BUSINESSCATEGORY|TELEPHONENUMBER|PSEUDONYM|POSTALADDRESS|NAME|DN|DESCRIPTION|(( [0-9]+\\. )+[0-9]+))=(?: (?:\\\\" | [^,\\\\" ])+) (?:, \\s? (?:EMAILADDRESS|CN|UID|SERIALNUMBER|OU|O|L|ST|C|T|SURNAME|STREET|INITIALS|GIVENNAME|DC|UNSTRUCTUREDADDRESS|UNSTRUCTUREDNAME|POSTALCODE|BUSINESSCATEGORY|TELEPHONENUMBER|PSEUDONYM|POSTALADDRESS|NAME|DN|DESCRIPTION|(( [0-9]+\\. )+[0-9]+))=(?: (?:\\\\" | [^,\\\\" ])+) )*)?*$",
        "defaultValue": null,
        "alert": null,
        "code": "DIRECTORY_NAME",
        "generalName": 4,
        "oid": null
    },
    {
        "index": 0,
        "name": "MS UPN, User Principal Name",
        "description": "MS UPN",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-zA-Яa-я0-9._\\/-]+@[A-Za-zA-Яa-я0-9._\\/-]+$",
        "defaultValue": null,
        "alert": null,
        "code": "MS_UPN",
        "generalName": 0,
        "oid": "1.3.6.1.4.1.311.20.2.3"
    },
    {

```

```

        "index": 0,
        "name": "MS GUID, Globally Unique Identifier",
        "description": "MS GUID",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Fa-f0-9]{32}$",
        "defaultValue": null,
        "alert": null,
        "code": "MS_GUID",
        "generalName": 0,
        "oid": "1.3.6.1.4.1.311.25.1"
    },
    {
        "index": 0,
        "name": "Kerberos KPN, Kerberos 5 Principal Name",
        "description": "Основное имя Kerberos 5",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-zA-Яа-я0-9._\\/-]+@[A-Za-zA-Яа-я0-9._\\/-]+$",
        "defaultValue": null,
        "alert": null,
        "code": "KRB5PRINCIPAL",
        "generalName": 0,
        "oid": "1.3.6.1.5.2.2"
    },
    {
        "index": 0,
        "name": "Xmpp address",
        "description": "Xmpp адрес",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^[A-Za-zA-Яа-я0-9._\\/-]+@[A-Za-zA-Яа-я0-9._\\/-]+$",
        "defaultValue": null,
        "alert": null,
        "code": "XMPP_ADDR",
        "generalName": 0,
        "oid": "1.3.6.1.5.5.7.8.5"
    },
    {
        "index": 0,
        "name": "Subject Identification Method",
        "description": "Метод идентификации субъекта",
        "required": false,
        "validation": true,
        "modifiable": true,
        "regex": "^( ([0-1]\\.[1-9]) | ([2]\\.[1-9]\\d*) | ([0-1]\\.[1-3]\\d) ) ( (\\.0) | (\\. [1-9]\\d*) ) * : : \\w+ : : \\w+$",
        "defaultValue": null,
        "alert": null,
        "code": "SUBJECT_IDENTIFICATION_METHOD",
        "generalName": 0,
        "oid": "1.3.6.1.5.5.7.8.6"
    }

```

```
    },  
    {  
      "index": 0,  
      "name": "Permanent identifier",  
      "description": "Постоянный идентификатор",  
      "required": false,  
      "validation": true,  
      "modifiable": true,  
      "regex": "^.*?\\./(([0-1]\\.[1-9])|([2]\\.[1-9]\\d*)|([0-1]\\.[1-  
3]\\d))((\\.0)|([1-9]\\d*))*$",  
      "defaultValue": null,  
      "alert": null,  
      "code": "PERMANENT_IDENTIFIER",  
      "generalName": 0,  
      "oid": "1.3.6.1.5.5.7.8.3"  
    }  
  ],  
  "created": "2024-07-30T17:04:45.593430Z",  
  "updated": null  
}
```

МЕТОДЫ РАБОТЫ С ЦЕНТРАМИ СЕРТИФИКАЦИИ

1.12 Метод получения активного ЦС

GET API – Получение активного ЦС	
Метод доступен администратору и оператору	
URL – certificate-authority-service/api/v2a/public/certificate-authorities/active	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a] Контроллер%3A Центры сертификации/active	
Query	
-	
Request	
-	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID ЦС
isActive (boolean),	Флаг: активный ЦС
active (boolean),	Флаг: активный ЦС
isManagement (boolean),	Флаг: технологический ЦС
management (boolean),	Флаг: технологический ЦС
certificate: {	Сертификат ЦС
id (UUID),	Идентификатор сертификата ЦС
issuerId (UUID),	Идентификатор издателя сертификата ЦС
issuerFingerprint (string),	Фингерпринт издателя сертификата ЦС
serialnumber (string),	Серийный номер сертификата ЦС
fingerprint (string),	Фингерпринт сертификата ЦС
issuerDN: {	Имя субъекта издателя сертификата ЦС
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectDN: {	Имя субъекта сертификата ЦС
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectAltName: {	Альтернативное имя субъекта сертификата ЦС
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
name (string),	Имя сертификата ЦС (на основе CN)
templateId (UUID),	Идентификатор шаблона
templateName (string),	Имя шаблона
validFrom (instant),	Дата начала действия сертификата ЦС (ISO 8601)
validTo (instant),	Дата окончания действия сертификата ЦС (ISO 8601)

status (enum: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN),	Статус сертификата ЦС
isValid (boolean),	Флаг: сертификат ЦС действует
isExpired (boolean),	Флаг: сертификат ЦС истек
actions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
revocation: {	Сведения об отзыве сертификата ЦС
date (instant),	Дата отзыва
number (int32),	Код причины отзыва
value (string)	Значение причины отзыва
},	
hashAlgorithm (enum: SHA1, SHA256, SHA384, SHA512, SHA3_256, SHA3_384, SHA3_512, RSASSA_PSS, MD5, GOST_R_34_11_2012, UNKNOWN),	Алгоритм подписи сертификата ЦС
keyAlgorithm (enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN),	Алгоритм ключа сертификата ЦС
keyBits (int32),	Длина ключа сертификата ЦС
subjectKeyIdentifier (string),	Идентификатор ключа сертификата ЦС
authorityKeyIdentifier (string),	Идентификатор ключа издателя сертификата ЦС
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
},	
chain: {	Цепочка сертификатов ЦС (рекурсивный объект)
id (UUID),	Идентификатор сертификата
name (string),	Имя сертификата (на основе CN)
subjectDN: {	Имя субъекта сертификата ЦС
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
issuer: {...}	Издатель сертификата (вложенный объект)
},	
certificateCount (int64),	Число выпущенных сертификатов
title (string),	Отображаемое имя ЦС
cryptographyProviders: {	Конфигурация криптопровайдеров алгоритмов ЦС
(enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN): {	Название алгоритма
cryptographyProvider (enum: DEFAULT, CRYPTO_PRO, UNKNOWN),	Название криптопровайдера
isAvailable (boolean),	Флаг: Доступность криптопровайдера
},	
primaryCryptographyProvider: {	Криптопровайдер закрытого ключа
cryptographyProvider (enum: DEFAULT, CRYPTO_PRO, UNKNOWN),	Название криптопровайдера
isAvailable (boolean),	Флаг: Доступность криптопровайдера
},	
isAvailable (boolean),	Флаг: Доступность ЦС
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	


```
    ],
    "O": [
        "orgn"
    ],
    "L": [
        "City"
    ],
    "DC": [
        "Comnt"
    ],
    "C": [
        "RU"
    ]
},
"subjectAltName": {},
"name": "aladdin",
"templateId": "9129245a-eaad-4ebc-a2a4-8845ac0336fb",
"templateName": "Root CA",
"validFrom": "2024-07-30T14:10:27Z",
"validTo": "2034-07-30T14:09:24Z",
"status": "ACTIVE",
"isValid": true,
"isExpired": false,
"actions": {
    "p12": false,
    "pem": true,
    "csr": false
},
"revocation": null,
"hashAlgorithm": "SHA256",
"keyAlgorithm": "RSA",
"keyBits": 1024,
"subjectKeyIdentifier": "51d47e459115bfa6aa567927ca48b6babe1319ae",
"authorityKeyIdentifier": "51d47e459115bfa6aa567927ca48b6babe1319ae",
"created": "2024-07-30T14:10:26.993807Z",
"updated": "2024-07-30T14:10:26.993817Z"
},
"chain": {
    "id": "da7ca68f-edaf-47a8-bdd9-cb08157f0b4b",
    "name": "aladdin",
    "subjectDN": {
        "CN": [
            "aladdin"
        ],
        "OU": [
            "Depnt"
        ],
        "O": [
            "orgn"
        ],
        "L": [
            "City"
        ],
        "DC": [
            "Comnt"
        ]
    }
}
```

```

        ],
        "C": [
            "RU"
        ]
    },
    "issuer": null
},
"certificateCount": 13,
"title": "Root999",
"cryptographyProviders": {
    "ECDSA": {
        "cryptographyProvider": "DEFAULT",
        "isAvailable": true
    },
    "RSA": {
        "cryptographyProvider": "DEFAULT",
        "isAvailable": true
    }
},
"primaryCryptographyProvider": {
    "cryptographyProvider": "DEFAULT",
    "isAvailable": true
},
"isAvailable": true,
"created": null,
"updated": null
}
}

```

1.13 Метод получения ЦС по идентификатору

GET API – Получение ЦС по идентификатору	
Метод доступен администратору	
URL – certificate-authority-service/api/v2a/public/certificate-authorities/{id}	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a	
Контроллер%3A Центры сертификации/findById_14	
Query	
{	
id (UUID)	ID ЦС
}	
Request	
-	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID ЦС
isActive (boolean),	Флаг: активный ЦС
active (boolean),	Флаг: активный ЦС
isManagement (boolean),	Флаг: технологический ЦС
management (boolean),	Флаг: технологический ЦС
certificate: {	Сертификат ЦС
id (UUID),	Идентификатор сертификата ЦС
issuerId (UUID),	Идентификатор издателя сертификата ЦС
issuerFingerprint (string),	Фингерпринт издателя сертификата ЦС
serialnumber (string),	Серийный номер сертификата ЦС
fingerprint (string),	Фингерпринт сертификата ЦС
issuerDN: {	Имя субъекта издателя сертификата ЦС

(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectDN: {	Имя субъекта сертификата ЦС
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectAltName: {	Альтернативное имя субъекта сертификата ЦС
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
name (string),	Имя сертификата ЦС (на основе CN)
templateId (UUID),	Идентификатор шаблона
templateName (string),	Имя шаблона
validFrom (instant),	Дата начала действия сертификата ЦС (ISO 8601)
validTo (instant),	Дата окончания действия сертификата ЦС (ISO 8601)
status (enum: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN),	Статус сертификата ЦС
isValid (boolean),	Флаг: сертификат ЦС действует
isExpired (boolean),	Флаг: сертификат ЦС истек
actions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
revocation: {	Сведения об отзыве сертификата ЦС
date (instant),	Дата отзыва
number (int32),	Код причины отзыва
value (string)	Значение причины отзыва
},	
hashAlgorithm (enum: SHA1, SHA256, SHA384, SHA512, SHA3_256, SHA3_384, SHA3_512, RSASSA_PSS, MD5, GOST_R_34_11_2012, UNKNOWN),	Алгоритм подписи сертификата ЦС
keyAlgorithm (enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN),	Алгоритм ключа сертификата ЦС
keyBits (int32),	Длина ключа сертификата ЦС
subjectKeyIdentifier (string),	Идентификатор ключа сертификата ЦС
authorityKeyIdentifier (string),	Идентификатор ключа издателя сертификата ЦС
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
},	
chain: {	Цепочка сертификатов ЦС (рекурсивный объект)
id (UUID),	Идентификатор сертификата
name (string),	Имя сертификата (на основе CN)
subjectDN: {	Имя субъекта сертификата ЦС

(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
issuer: {...}	Издатель сертификата (вложенный объект)
},	
certificateCount (int64),	Число выпущенных сертификатов
title (string),	Отображаемое имя ЦС
cryptographyProviders: {	Конфигурация криптопровайдеров алгоритмов ЦС
(enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN): {	Название алгоритма
cryptographyProvider (enum: DEFAULT, CRYPTO_PRO, UNKNOWN),	Название криптопровайдера
isAvailable (boolean),	Флаг: Доступность криптопровайдера
},	
primaryCryptographyProvider: {	Криптопровайдер закрытого ключа
cryptographyProvider (enum: DEFAULT, CRYPTO_PRO, UNKNOWN),	Название криптопровайдера
isAvailable (boolean),	Флаг: Доступность криптопровайдера
},	
isAvailable (boolean),	Флаг: Доступность ЦС
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.13.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/certificate-authority-
service/api/v2a/public/certificate-authorities/45f34462-62a4-4671-9927-
f1c6763a4470' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI2YjNiNTRkZS01YjU2LTQzMzktOTZmMy0lZWElZWfknjAxMTMiLCJpYXQiOiJE3MDIyNzc5MzgsImV4cCI6MTcwMjI3ODExOH0.Qpz1lAH4pDr
2953zYv06fq9djAwoAePDwg84FPyj9oVIyNIvpgg2-frW9X8s7w316cczPBWXUdwaNxpDnpirpy6u-
O3j43CLcuGFhoBN7B2Dpsx_L7UTmoo-
Eq1Uk7Ld1HBEYcc7EX8GBU2598CwW42Ao5gK49w1fw13zcl1h7u_Wlq1PupI1_XckGnnmqkF9F8lX2-
WJ2Ld2H3Jp5fbe81MlOllsxB-
WtJUxmQvAL0N4Ogr6ucOIRgUH9HJ1WekNRD1yuvHfCjqLsssbVikrO_nmu4sAKaExAaZWix8-
e59h88xfN4uONQWWq55iHktiHQFP2YM3W3RYth9vQNRA'
```

1.13.2 Пример ответа на обращение к методу

```
{
  "status": 200,
  "data": {
    "id": "45f34462-62a4-4671-9927-f1c6763a4470",
    "isActive": true,
    "active": true,
    "isManagement": false,
    "management": false,
    "certificate": {
      "id": "da7ca68f-edaf-47a8-bdd9-cb08157f0b4b",
      "issuerId": "da7ca68f-edaf-47a8-bdd9-cb08157f0b4b",
      "issuerFingerprint": "10faca35d9763c66aa07be57d844092a11ac1040",
      "serialnumber": "540ad4f920bd7ae95464ede5c03a3ca4147ce4f0",
```

```
"fingerprint": "10faca35d9763c66aa07be57d844092a11ac1040",
"issuerDN": {
  "CN": [
    "aladdin"
  ],
  "OU": [
    "Depnt"
  ],
  "O": [
    "orgn"
  ],
  "L": [
    "City"
  ],
  "DC": [
    "Comnt"
  ],
  "C": [
    "RU"
  ]
},
"subjectDN": {
  "CN": [
    "aladdin"
  ],
  "OU": [
    "Depnt"
  ],
  "O": [
    "orgn"
  ],
  "L": [
    "City"
  ],
  "DC": [
    "Comnt"
  ],
  "C": [
    "RU"
  ]
},
"subjectAltName": {},
"name": "aladdin",
"templateId": "9129245a-eaad-4ebc-a2a4-8845ac0336fb",
"templateName": "Root CA",
"validFrom": "2024-07-30T14:10:27Z",
"validTo": "2034-07-30T14:09:24Z",
"status": "ACTIVE",
"isValid": true,
"isExpired": false,
"actions": {
  "p12": false,
  "pem": true,
  "csr": false
},

```

```
    "revocation": null,
    "hashAlgorithm": "SHA256",
    "keyAlgorithm": "RSA",
    "keyBits": 1024,
    "subjectKeyIdentifier": "51d47e459115bfa6aa567927ca48b6babe1319ae",
    "authorityKeyIdentifier": "51d47e459115bfa6aa567927ca48b6babe1319ae",
    "created": "2024-07-30T14:10:26.993807Z",
    "updated": "2024-07-30T14:10:26.993817Z"
  },
  "chain": {
    "id": "da7ca68f-edaf-47a8-bdd9-cb08157f0b4b",
    "name": "aladdin",
    "subjectDN": {
      "CN": [
        "aladdin"
      ],
      "OU": [
        "Depnt"
      ],
      "O": [
        "orgn"
      ],
      "L": [
        "City"
      ],
      "DC": [
        "Comnt"
      ],
      "C": [
        "RU"
      ]
    },
    "issuer": null
  },
  "certificateCount": 13,
  "title": "Root999",
  "cryptographyProviders": {
    "ECDSA": {
      "cryptographyProvider": "DEFAULT",
      "isAvailable": true
    },
    "RSA": {
      "cryptographyProvider": "DEFAULT",
      "isAvailable": true
    }
  },
  "primaryCryptographyProvider": {
    "cryptographyProvider": "DEFAULT",
    "isAvailable": true
  },
  "isAvailable": true,
  "created": null,
  "updated": null
}
```

1.14 Метод получения Центров сертификации

GET API – Получение ЦС	
Метод доступен администратору	
URL – certificate-authority-service/api/v2a/public/certificate-authorities	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a	
Контроллер%3A Центры сертификации/findAll_15	
Query	
{	
status (enum[:ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN) [опционально],	Статус сертификата ЦС
type (enum[: CERTIFICATE ² , ROOT_CA, SUB_CA, UNKNOWN) [опционально],	Тип сертификата ЦС
search (string) [опционально],	Полнотекстовый поиск по имени ЦС
isManagement (boolean) [опционально],	Флаг: технологический ЦС
isActive (boolean) [опционально],	Флаг: активный ЦС
isValid (boolean) [опционально],	Флаг: сертификат ЦС действителен
isExpired (boolean) [опционально],	Флаг: сертификат ЦС истек
id (UUID[]) [опционально],	ID ЦС
notIds (UUID[]) [опционально],	Исключая ID ЦС
sortDirection (string) [опционально],	Направления сортировки (ASC;DESC)
sortBy (string[]) [опционально],	Список полей, к которым применяется сортировка
pageOffset (integer) [опционально],	Смещение от начала списка (пагинация)
pageLimit (integer) [опционально]	Ограничение на размер выборки (пагинация)
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID ЦС
isActive (boolean),	Флаг: активный ЦС
active (boolean),	Флаг: активный ЦС
isManagement (boolean),	Флаг: технологический ЦС
management (boolean),	Флаг: технологический ЦС
certificate: {	Сертификат ЦС
id (UUID),	Идентификатор сертификата ЦС
issuerId (UUID),	Идентификатор издателя сертификата ЦС
issuerFingerprint (string),	Фингерпринт издателя сертификата ЦС
serialnumber (string),	Серийный номер сертификата ЦС
fingerprint (string),	Фингерпринт сертификата ЦС
issuerDN: {	Имя субъекта издателя сертификата ЦС
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectDN: {	Имя субъекта сертификата ЦС
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE,	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра

2 Тип `CERTIFICATE` является общим для всех словарей типов сертификатов в программе. При использовании данного метода указание данного типа также доступно, однако сертификаты ЦС с данным типом отсутствуют, соответственно не будут найдены и возвращены в ответе.

BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	
},	
subjectAltName: {	Альтернативное имя субъекта сертификата ЦС
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
name (string),	Имя сертификата ЦС (на основе CN)
templateId (UUID),	Идентификатор шаблона
templateName (string),	Имя шаблона
validFrom (instant),	Дата начала действия сертификата ЦС (ISO 8601)
validTo (instant),	Дата окончания действия сертификата ЦС (ISO 8601)
status (enum: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN),	Статус сертификата ЦС
isValid (boolean),	Флаг: сертификат ЦС действует
isExpired (boolean),	Флаг: сертификат ЦС истек
actions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
revocation: {	Сведения об отзыве сертификата ЦС
date (instant),	Дата отзыва
number (int32),	Код причины отзыва
value (string)	Значение причины отзыва
},	
hashAlgorithm (enum: SHA1, SHA256, SHA384, SHA512, SHA3_256, SHA3_384, SHA3_512, RSASSA_PSS, MD5, GOST_R_34_11_2012, UNKNOWN),	Алгоритм подписи сертификата ЦС
keyAlgorithm (enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN),	Алгоритм ключа сертификата ЦС
keyBits (int32),	Длина ключа сертификата ЦС
subjectKeyIdentifier (string),	Идентификатор ключа сертификата ЦС
authorityKeyIdentifier (string),	Идентификатор ключа издателя сертификата ЦС
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
},	
chain: {	Цепочка сертификатов ЦС (рекурсивный объект)
id (UUID),	Идентификатор сертификата
name (string),	Имя сертификата (на основе CN)
subjectDN: {	Имя субъекта сертификата ЦС
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
issuer: {...}	Издатель сертификата (вложенный объект)
},	
certificateCount (int64),	Число выпущенных сертификатов
title (string),	Отображаемое имя ЦС
cryptographyProviders: {	Конфигурация криптопровайдеров алгоритмов ЦС

(enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN): {	Название алгоритма
cryptographyProvider (enum: DEFAULT, CRYPTO_PRO, UNKNOWN),	Название криптопровайдера
isAvailable (boolean),	Флаг: Доступность криптопровайдера
},	
primaryCryptographyProvider: {	Криптопровайдер закрытого ключа
cryptographyProvider (enum: DEFAULT, CRYPTO_PRO, UNKNOWN),	Название криптопровайдера
isAvailable (boolean),	Флаг: Доступность криптопровайдера
},	
isAvailable (boolean),	Флаг: Доступность ЦС
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.14.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/certificate-authority-
service/api/v2a/public/certificate-authorities' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI2YjNiNTRkZS01YjU2LTQzMz
ktOTZmMy01ZWElZWFKNjAxMTMiLCJpYXQiOiE3MDIyNzc5MzgsImV4cCI6MTcwMjI3ODExOH0.Qpz1lAH4pDr
2953zYv06fq9djAwoAePDwg84FPyj9oVIyNIvpgg2-frW9X8s7w316cczPBWXUdwaNxpDnpirpy6u-
O3j43CLcuGFhoBN7B2Dpsx_L7UTmoo-
Eq1Uk7Ld1HBEYCC7EX8GBU2598CwW42Ao5gK49w1fw13zcliH7u_Wlq1PupI1_XckGnnmqkF9F8lX2-
WJ2Ld2H3Jp5fbe81MlOllsxB-
WtJUxmQvAL0N4Ogr6ucOIRgUH9HJ1WekNRD1yuvHfCjQlsssbVikrO_nmu4sAKaExAaZWix8-
e59h88xfN4uONQWWq55iHktiHQFP2YM3W3RYth9vQNRA'
```

1.14.2 Пример ответа на обращение к методу

```
{
  "status": 200,
  "data": {
    "range": {
      "count": 2,
      "offset": 0,
      "limit": 10,
      "totalPages": 1,
      "pageNumber": 0
    },
    "items": [
      {
        "id": "80c6da0a-9a97-4269-99fe-ed52615034f1",
        "active": false,
        "isActive": false,
        "management": true,
        "isManagement": true,
        "certificate": {
          "id": "4f5f6412-7cc9-48a7-ba48-2e9e9e76da37",
          "issuerId": "4f5f6412-7cc9-48a7-ba48-2e9e9e76da37",
          "issuerFingerprint": "165ce20a52590dd8a496c1f1c0b5cf0812728de7",
          "serialnumber": "25988edc904cbf3244059d327d103538a910ad38",
          "fingerprint": "165ce20a52590dd8a496c1f1c0b5cf0812728de7",
          "issuerDN": {
            "CN": [
```

```
        "INITIAL_CA"
      ],
    },
    "subjectDN": {
      "CN": [
        "INITIAL_CA"
      ]
    },
    "subjectAltName": {},
    "name": "INITIAL_CA",
    "templateId": "9129245a-eaad-4ebc-a2a4-8845ac0336fb",
    "templateName": "Root CA",
    "validFrom": "2024-07-24T13:23:56Z",
    "validTo": "2048-07-25T13:23:56Z",
    "status": "ACTIVE",
    "isValid": true,
    "isExpired": false,
    "actions": {
      "p12": false,
      "pem": true,
      "csr": false
    },
    "revocation": null,
    "hashAlgorithm": "SHA256",
    "keyAlgorithm": "RSA",
    "keyBits": 2048,
    "subjectKeyIdentifier":
"fe4afd831276234c8fc511a86093f3fcf647acac",
    "authorityKeyIdentifier":
"fe4afd831276234c8fc511a86093f3fcf647acac",
    "created": "2024-07-24T13:23:57.229542Z",
    "updated": "2024-07-24T13:23:57.229581Z"
  },
  "chain": {
    "id": "4f5f6412-7cc9-48a7-ba48-2e9e9e76da37",
    "name": "INITIAL_CA",
    "subjectDN": {
      "CN": [
        "INITIAL_CA"
      ]
    },
    "issuer": null
  },
  "certificateCount": 2,
  "title": "INITIAL_CA",
  "cryptographyProviders": {
    "RSA": {
      "cryptographyProvider": "DEFAULT",
      "isAvailable": true
    },
    "ECDSA": {
      "cryptographyProvider": "DEFAULT",
      "isAvailable": true
    }
  }
},
```

```
    "primaryCryptographyProvider": {
      "cryptographyProvider": "DEFAULT",
      "isAvailable": true
    },
    "isAvailable": true,
    "created": null,
    "updated": null
  },
  {
    "id": "aa7028d8-80dd-4665-bca3-d8b36f53fa1b",
    "active": true,
    "isActive": true,
    "management": false,
    "isManagement": false,
    "certificate": {
      "id": "33f7d408-82e1-405c-a5d0-f06cbeb10612",
      "issuerId": "33f7d408-82e1-405c-a5d0-f06cbeb10612",
      "issuerFingerprint": "1a438a200ca4192b299c954789cce55395ab3daf",
      "serialnumber": "5ab6dd5722ab2ce42c4f53910c771765a04c2376",
      "fingerprint": "1a438a200ca4192b299c954789cce55395ab3daf",
      "issuerDN": {
        "CN": [
          "aladdin"
        ],
        "OU": [
          "Deparnt"
        ],
        "O": [
          "orgon"
        ],
        "L": [
          "City"
        ],
        "DC": [
          "Cnent"
        ],
        "C": [
          "RU"
        ]
      },
      "subjectDN": {
        "CN": [
          "aladdin"
        ],
        "OU": [
          "Deparnt"
        ],
        "O": [
          "orgon"
        ],
        "L": [
          "City"
        ],
        "DC": [
          "Cnent"
        ]
      }
    }
  }
}
```

```
    ],
    "C": [
        "RU"
    ]
},
"subjectAltName": {},
"name": "aladdin",
"templateId": "9129245a-eaad-4ebc-a2a4-8845ac0336fb",
"templateName": "Root CA",
"validFrom": "2024-07-24T13:27:53Z",
"validTo": "2034-07-24T13:26:58Z",
"status": "ACTIVE",
"isValid": true,
"isExpired": false,
"actions": {
    "p12": false,
    "pem": true,
    "csr": false
},
"revocation": null,
"hashAlgorithm": "SHA256",
"keyAlgorithm": "RSA",
"keyBits": 4096,
"subjectKeyIdentifier":
"f047a6659c36e72bc22alc2eb2ec1971ff79b857",
"authorityKeyIdentifier":
"f047a6659c36e72bc22alc2eb2ec1971ff79b857",
"created": "2024-07-24T13:27:52.869628Z",
"updated": "2024-07-24T13:27:52.869640Z"
},
"chain": {
    "id": "33f7d408-82e1-405c-a5d0-f06cbeb10612",
    "name": "aladdin",
    "subjectDN": {
        "CN": [
            "aladdin"
        ],
        "OU": [
            "Deparnt"
        ],
        "O": [
            "orgon"
        ],
        "L": [
            "City"
        ],
        "DC": [
            "Cnent"
        ],
        "C": [
            "RU"
        ]
    },
    "issuer": null
},
```

```
    "certificateCount": 3,  
    "title": "Root15",  
    "cryptographyProviders": {  
      "RSA": {  
        "cryptographyProvider": "DEFAULT",  
        "isAvailable": true  
      },  
      "ECDSA": {  
        "cryptographyProvider": "DEFAULT",  
        "isAvailable": true  
      }  
    },  
    "primaryCryptographyProvider": {  
      "cryptographyProvider": "DEFAULT",  
      "isAvailable": true  
    },  
    "isAvailable": true,  
    "created": null,  
    "updated": null  
  }  
]  
}
```

МЕТОДЫ РАБОТЫ С СЕРТИФИКАТАМИ

1.15 Метод выпуска сертификата в контейнере pkcs#12

POST API – Выпуск сертификата в контейнере pkcs#12	
Метод доступен: – администратору; – оператору при наличии полномочий на управление субъектами и использование шаблона, идентификаторы которых передаются во входных параметрах. Использование данного метода оператором для создания сертификатов для учетных записей должно быть запрещено.	
URL – certificate-authority-service/api/v2a/public/certificates/enroll/{cald}	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A сертификаты/enrollByCald	
Query	
{	
caId (UUID),	ID ЦС
subjectId (UUID) [обязателен, если не указан userId],	ID субъекта
userId (UUID) [обязателен, если не указан subjectId]	ID учетной записи
}	
Request	
{	
templateId (UUID),	Идентификатор шаблона ³
subjectDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] },	Поля разделенного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] },	Поля альтернативного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
keyBits (integer),	Длина ключа
keyAlgorithm (enum: RSA, ECDSA, GOST R 34 10 2012, UNKNOWN),	Алгоритм ключевой пары сертификата
password (string)	Пароль контейнера
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID сертификата
downloadActions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10

³ Шаблоны в eCA-CA 2.2 содержат поле «Центр сертификации» (поле «certificateAuthorityId»), определяющее ЦС, на котором должен быть издан сертификат. В случае, если для указанного в поле «templateId» шаблона задан ЦС, отличный от указанного в поле «cald», ответ данного метода будет иметь код 500 и будет содержать сообщение об ошибке «Шаблон {идентификатор шаблона} не может быть использован для выпуска сертификата на центре сертификации {идентификатор центра сертификации из поля «cald»}. При использовании шаблона, в котором в поле «Центр сертификации» указано значение «Любой», выпуск сертификата будет происходить на ЦС, указанном в поле «cald».

pem (boolean)	Флаг: выгрузка сертификата
},	
fingerprint (string),	Фингерпринт шаблона
serialnumber (string),	Серийный номер сертификата
templateId (UUID),	ID шаблона
templateName (string),	Имя шаблона
name (string),	Имя сертификата (на основе CN)
issuerDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] },	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
subjectDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] },	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] },	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant)	Дата окончания действия сертификата (ISO 8601)
}	

1.15.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/certificate-authority-
service/api/v2a/public/certificates/enroll/7efd4f36-29ef-481e-ad0b-
2dc143d9807c?subjectId=000000a6-d0c3-4a4e-ae6-be5ad83a3ba2\
--header 'Content-Type: application/json' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI2YmRiMjQ2MS03MGY5LTZiZD
ItODA2NC0yYjgwM2M4OWY4ZTciLCJpYXQiOiE3MDIzMDU3NzYsImV4cCI6MTcwMjMwNTk1Nn0.ItIqTdQWF4c
hiljlxjHNYyYaYUyO34-bZ-3PADuR_f68McSjF6c-VwWIOUkQC1MG9EhA0MOX1AD1hUMLc-
Kd8vfl7gRRuQUiIhF4ii55D815f3z2dlrtkYc8BHioFwUrMvyCBo7m3CnD17VQ_IOAxFZjLwQendelwrJs6OG
PusmST21_kBNdXocfs5SJ3wzxBIITYTw94pM_onvQzHo8J30CBqUcwihQ5Gr5N9VQ5o7oCXjwonxbJaYi5eRK
UfQdae9weUFSeq-
cFTQwTF2CmGKru2uHAnea66sNaFHLXadAqjW_DrPupNXXjuB0eWtsq2ljYzyTw2M31pbAdm3brQ' \
--data-raw '{
"templateId":"059a38f5-f345-4275-b79f-e7e6cc3cbb68",
"subjectDN":{
"CN":["Dena Marshall"]
},
"subjectAltName":{
"RFC822NAME":["Thenterage87@test.local"],
"MS_UPN":["Thenterage87@test.local"]
},
}
```

```
"keyBits":2048,  
"keyAlgorithm":"RSA",  
"password":"werwer1QQ"  
}
```

1.15.2 Пример ответа на обращение к методу

```
{  
  "status": 200,  
  "data": {  
    "id": "7bde3592-e092-4f8b-ad1c-5ba3323ebbab",  
    "downloadActions": {  
      "p12": true,  
      "pem": true,  
      "csr": false  
    },  
    "fingerprint": "692297d8597376406ded21d1ae5cb4692a00b74e",  
    "serialNumber": "152aaa67cc9c67b8a27c77e0f457a742847ced2c",  
    "templateId": "059a38f5-f345-4275-b79f-e7e6cc3cbb68",  
    "templateName": "WEB-Client",  
    "name": "Dena Marshall",  
    "issuerDN": {  
      "CN": [  
        "SUB_CA_INFORM"  
      ],  
      "OU": [  
        "Department"  
      ],  
      "O": [  
        "organization"  
      ],  
      "L": [  
        "City"  
      ],  
      "C": [  
        "RU"  
      ],  
      "DC": [  
        "Component"  
      ]  
    },  
    "subjectDN": {  
      "CN": [  
        "Dena Marshall"  
      ]  
    },  
    "subjectAltName": {  
      "RFC822NAME": [  
        "Thenterage87@test.local"  
      ],  
      "MS_UPN": [  
        "Thenterage87@test.local"  
      ]  
    },  
    "validFrom": "2024-02-06T12:24:33.522558492Z",  
  }  
}
```

```

    "validTo": "2026-02-05T12:24:33.522558492Z"
  }
}

```

1.16 Методы выпуска сертификата по запросу pkcs#10

1.16.1 Выпуск сертификата по запросу pkcs#10 (формат запроса на сертификат из входных параметров – multipart/form-data)

POST API – Выпуск сертификата по запросу pkcs#10 (multipart/form-data)	
Метод доступен: – администратору; – оператору при наличии полномочий на управление субъектами и использование шаблона, идентификаторы которых передаются во входных параметрах. Использование данного метода оператором для создания сертификатов для учетных записей должно быть запрещено.	
URL – certificate-authority-service/api/v2a/public/certificates/enroll/{cald}/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2aКонтроллер%3A сертификаты/enrollRequestByCald_1	
Query	
{	
caId (UUID),	ID ЦС
subjectId (UUID) [обязателен, если не указан userId],	ID субъекта
userId (UUID) [обязателен, если не указан subjectId]	ID учетной записи
}	
Request	
{	
templateId (UUID),	Идентификатор шаблона ⁴
request (binary),	Файл запроса на сертификат (см. пример использования метода ниже)
subjectName: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] }, [опционально]	Поля отличительного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectName, переопределяют значения соответствующих полей SDN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] } [опционально]	Поля альтернативного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectAltName, переопределяют значения соответствующих полей SAN запроса на сертификат, при

⁴ Шаблоны в eCA-CA 2.2 содержат поле «Центр сертификации» (поле «certificateAuthorityId»), определяющее ЦС, на котором должен быть издан сертификат. В случае, если для указанного в поле «templateId» шаблона задан ЦС, отличный от указанного в поле «cald», ответ данного метода будет иметь код 500 и будет содержать сообщение об ошибке «Шаблон {идентификатор шаблона} не может быть использован для выпуска сертификата на центре сертификации {идентификатор центра сертификации из поля «cald»}. При использовании шаблона, в котором в поле «Центр сертификации» указано значение «Любой», выпуск сертификата будет происходить на ЦС, указанном в поле «cald».

	условии, что они соответствуют значениям атрибутов субъекта.
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID сертификата
downloadActions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
fingerprint (string),	Фингерпринт шаблона
serialnumber (string),	Серийный номер сертификата
templateId (UUID),	ID шаблона
templateName (string),	Имя шаблона
name (string),	Имя сертификата (на основе CN)
issuerDN: {	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	
},	
subjectDN: {	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	
},	
subjectAltName: {	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	
},	
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant)	Дата окончания действия сертификата (ISO 8601)
}	

1.16.1.1 Пример использования метода через Curl:

```
curl -k --request POST --location 'https://192.168.111.18/certificate-authority-  
service/api/v2a/public/certificates/enroll/e7e2fdac-b863-4194-9174-  
bed0a67f25e8/pkcs10?subjectId=48a4b6dd-a5e9-45eb-ad0c-59923618dc4c' \  
--header 'Cookie:  
token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI0ZmFhYjgxNi0wZGNhLTQ0ZjctOGQzZS0xNDkwNTdmYzYwZmEiLCJpYXQiOiJlM0IiwiaWF0IjE2MTcwMjAyODE2Nn0.EBQw9zAD-  
I8BsCkYT5A8mUdyjBc7f14Ogm17ui0IdjpdK6Eq4k5UQ13Vgj2kMhU2XQKnbt4NZNJ45Y8d-  
_FXXwk_1KcG94X8YvrmF6tWHnAOt3DLjsIMm3ZbMJjAopQx-0PQURg8_aQ6d4E2rLWB-  
mWE1h8JqcwZ5g29yD1qHS7V0ArSy_AbkFmnqUrrq6CVCQLETGef05qCIIX5_WEB9bF-  
bnevA9aqlTPWdYzbcXTAUgRumVKEKgi_if9S-  
mSYFC8EtYvztC30klEtssR6wG2VucIozLhYRuWVavXD70XyQ8pRtKxYIQimt7cuD5Ez7v-  
ML9LfDZ5B6Eih1OM7g' \  
--form 'request=@"C:/Users/user/Downloads/request.csr"'
```

1.16.2 Выпуск сертификата по запросу pkcs#10 (формат запроса на сертификат из входных параметров – application/json)

Примечание: отличается от метода выше форматом передаваемого во входных параметрах запроса на сертификат. В данном методе на вход поступает содержимое файла запроса на сертификат в формате PEM (содержимое в Base64).

POST API – Выпуск сертификата по запросу pkcs#10 (application/json)	
Метод доступен:	
– администратору; – оператору при наличии полномочий на управление субъектами и использование шаблона, идентификаторы которых передаются во входных параметрах.	
Использование данного метода оператором для создания сертификатов для учетных записей должно быть запрещено.	
URL – certificate-authority-service/api/v2a/public/certificates/enroll/{cald}/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A%20сертификаты/enrollRequestByCald_1	
Query	
{	
caId (UUID),	ID ЦС
subjectId (UUID) [обязателен, если не указан userId],	ID субъекта
userId (UUID) [обязателен, если не указан subjectId]	ID учетной записи
}	
Request	
{	
templateId (UUID),	Идентификатор шаблона ⁵
request: {	Запрос на сертификат
contentType(string) [опционально],	Тип загружаемого файла (HTTP MediaType) – application/octet-stream
fileName (string) [опционально],	Имя загружаемого файла
data (string:binary)	Содержимое PEM файла запроса на сертификат (массив байт в Base64) – см. пример использования метода ниже
},	
subjectName: {	Поля отличительного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра.
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Значения полей, указанные в subjectName, переопределяют значения соответствующих полей SDN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
}, [опционально]	
subjectAltName: {	Поля альтернативного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра.
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME,	

⁵ Шаблоны в eCA-CA 2.2 содержат поле «Центр сертификации» (поле «certificateAuthorityId»), определяющее ЦС, на котором должен быть издан сертификат. В случае, если для указанного в поле «templateId» шаблона задан ЦС, отличный от указанного в поле «cald», ответ данного метода будет иметь код 500 и будет содержать сообщение об ошибке «Шаблон {идентификатор шаблона} не может быть использован для выпуска сертификата на центре сертификации {идентификатор центра сертификации из поля «cald»}. При использовании шаблона, в котором в поле «Центр сертификации» указано значение «Любой», выпуск сертификата будет происходить на ЦС, указанном в поле «cald».

SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] } [опционально]	Значения полей, указанные в subjectAltName, переопределяют значения соответствующих полей SAN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID сертификата
downloadActions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
fingerprint (string),	Фингерпринт шаблона
serialnumber (string),	Серийный номер сертификата
templateId (UUID),	ID шаблона
templateName (string),	Имя шаблона
name (string),	Имя сертификата (на основе CN)
issuerDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] },	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
subjectDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] },	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] },	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant)	Дата окончания действия сертификата (ISO 8601)
}	

1.16.2.1 Пример использования метода через Curl:

```
curl -k --request POST --location 'https://192.168.111.18/certificate-authority-service/api/v2a/public/certificates/enroll/e7e2fdac-b863-4194-9174-bed0a67f25e8/pkcs10?subjectId=48a4b6dd-a5e9-45eb-ad0c-59923618dc4c' \
--header 'Content-Type: application/json' \
--data '{
  "templateId": "48a4b6dd-a5e9-45eb-ad0c-59923618dc4c",
  "request": {
    "contentType": "application/octet-stream",
    "fileName": "test",
```

```

    "data":
    "LS0tLS1CRUdJTiB0R2Vuc2Q0VSVElGSUNBVEUgUkVRVUUVTV0tLS0tCk1JSUNnekNDQVdzQ0FRQXdQakVQTUEw
R0ExVUVCaE1HVW5WemMybGhNUTR3REFZRFZRUUtdF1YwYUkKbFpURU0NCk1Bb0dBMVVFQ3hNRGRIZHZNUTB3Q
3dZRFZRUURFd1JTVDA5VU1JSUJJaFQmdrcWhraUc5dzBCQVFFRkFBT0MNCkFROEFNSU1CQ2dLQ0FRRUFpOV
RJeXRSck1FYi9kU1RhSWtxcEdZSGNkTlprb0Vnk2E2L090SVk3d3JCY2M2cEYnNCR1SudGc0VkrmpNcWpKbFB
1WmFhQ21EMDhVMU1td0h2VXBqK3AzSEd3YWdIRkk2elNrcnNEUTJnKzhDL1B0ZnENCnVqcU1vRUhNTWZJK3FE
TEdVemp6eGJUV0F1T05DVjBtUmNpdTlIU2o0Ynh4dTVUeW05QnNhYVBvczR6MzAycC8NCk13b3dlemxhCZ0EzV
2NYS1YzVTRSBw1VcERmcmlNNDRIWEw5R3BTejNJVkd1M1kxQWtlc3RpQXNCEjhwMUNTbU8NC1grT3hwUUVIMV
1FVHVSRFp0dGtTM2kvVmthV0JtTTZXK1VJWDZRYUwxTTdRSDFMTWVvY1ludlUwSEVMQjRwYysNCn1JN1dKK2t
Cc3JLKL1RNbHEyZfV5RTdnWFR4YUzks1lpSmZKTnVRSURBUUFcb0FBd0RRWUpLb1pJaHZjTkFRRUYNckJRQURn
Z0VCQUIwWHNGYzhYUGJmME9hVGdCVkK2K0R1RES5NzVBVHREZHRCRTEwBDYzdlcwZkhzZmM0WHJ5b1oNCkpaT
XRmaXRGUExxcGtaT1lFbEtYSGpiZlJzYzd2NzJSRmszbGpUQ1l0Ym9lNmFkT1IvWEVYMH1VS1pPM1lTNSsNCn
1NbE5neERNZ0NzeEhZQ1FoNzh0Yk1NUWIrSDhXZm5ZzF1L01sL3JmOVhQbENCL05Wdm5raS8wOHQvQ0dySVc
NCjZqb0R0bnhQcFdyYWdQUjV5c3I0Q1J6L01DMkU0cEFESjZQdVlkZW9EY1FUTE1yMGFTRG5NTGdNZGN4eFVL
NUkNCjNFb01zTEclDUpoZWVQODBESFFuYnVlbStjTU4rK31PcUxHa0FaaHdRGREano0cTAYelV1RnhpeEJBV
zcrRzYnCMkvejZhemRtTm5MR1VPSGZpYm5qN3Q2aFVKZUsxa289Ci0tLS0tRU5EIE5FVyBDRVJUSUZJQ0FURS
BSRVFVRVNULS0tLS0K"
  }
}'
--cookie
'token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiJlZjI0ODYyMiJzZjRlLTRmZmUtYTJkZi1mODdmMm
mViOGRkZTYiLCJpYXQiOiJlZjI0ODYyMiJzZjRlLTRmZmUtYTJkZi1mODdmMmViOGRkZTYiLCJpYXQiOiJlZjI0ODYyMiJzZjRlLTRmZmUtYTJkZi1mODdmMm
EObNpv2IV8I991m2hJrz6VmtQtzdyKihG7BDUZp_zOf_j63UbRmhtue59o6WQLii7sBRaFakckzxCP98mtc5m
uB9E77b1fu0OHcoZqZfSX36bSoH6R0hLZ1FyTzjRnSfdr1KisQAaF0XrjK5nSfmgNEQOCinAu2f82tbFtXfyS
1AeGjrlSL6A7EzQ_UvJNfFg4FWWyzgruSyJlVKAyn_kwdolJe1j_pOc4yr_rXH391IqX6GL7t1C5cUSaDCXwg
-hE49qkw9f9-2oBlip92Pug6dbeAPvPF5nnhH3JHw-3JogsrUsimTruqfl9GA'

```

1.17 Методы перевыпуска сертификата по запросу pkcs#10

1.17.1 Перевыпуск сертификата по запросу pkcs#10 (формат запроса на сертификат из входных параметров – multipart/form-data)

POST API – Перевыпуск сертификата по запросу pkcs#10 (multipart/form-data)
Метод доступен: – администратору; – оператору при наличии полномочий на управление субъектами и использование шаблона, идентификаторы которых передаются во входных параметрах. Использование данного метода оператором для создания сертификатов для учетных записей должно быть запрещено.
Метод позволяет перевыпустить сертификат по запросу, по которому ранее уже был выпущен сертификат. При использовании метода проверяется наличие в базе данных программы выпущенного сертификата, имеющего «Subject Key Identifier» аналогичный указанному в запросе на сертификат из входных параметров. Если такой сертификат не будет найден, данный метод осуществит выпуск нового сертификата по запросу аналогично методу выпуска сертификата по запросу (см. раздел 1.16). Если сертификат с аналогичным указанному в запросе на сертификат «Subject Key Identifier» будет найден, программа: 1) проверит статус найденного сертификата. Если срок действия данного сертификата истек, запрос пользователя будет отклонен.

2) проверит соответствие значений в полях SDN и SAN, указанных в запросе (или во входных параметрах метода), значениям в полях SDN и SAN в найденном сертификате. При соответствии значений будет осуществлен выпуск сертификата по запросу на сертификат из входных параметров метода, иначе запрос пользователя будет отклонен.	
URL – certificate-authority-service/api/v2a/public/certificates/renewal/{cald}/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A сертификаты/renewalRequestByCald_1	
Query	
{	
caId (UUID),	ID ЦС
subjectId (UUID) [обязателен, если не указан userId],	ID субъекта
userId (UUID) [обязателен, если не указан subjectId]	ID учетной записи
}	
Request	
{	
templateId (UUID),	Идентификатор шаблона
request (binary),	Файл запроса на сертификат
subjectName: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] }, [опционально]	Поля отличительного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectName, переопределяют значения соответствующих полей SDN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] } [опционально]	Поля альтернативного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectAltName, переопределяют значения соответствующих полей SAN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID сертификата
downloadActions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
fingerprint (string),	Фингерпринт шаблона
serialnumber (string),	Серийный номер сертификата
templateId (UUID),	ID шаблона
templateName (string),	Имя шаблона
name (string),	Имя сертификата (на основе CN)
issuerDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET,	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра

NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] },	
subjectDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] },	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] },	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant)	Дата окончания действия сертификата (ISO 8601)
}	

1.17.2 Перевыпуск сертификата по запросу pkcs#10 (формат запроса на сертификат из входных параметров – application/json)

Примечание: отличается от метода выше форматом передаваемого во входных параметрах запроса на сертификат. В данном методе на вход поступает содержимое файла запроса на сертификат в формате PEM (содержимое в Base64).

POST API – Перевыпуск сертификата в по запросу pkcs#10 (application/json)
Метод доступен: – администратору; – оператору при наличии полномочий на управление субъектами и использование шаблона, идентификаторы которых передаются во входных параметрах. Использование данного метода оператором для создания сертификатов для учетных записей должно быть запрещено.
Метод позволяет перевыпустить сертификат по запросу, по которому ранее уже был выпущен сертификат. При использовании метода проверяется наличие в базе данных программы выпущенного сертификата, имеющего «Subject Key Identifier» аналогичный указанному в запросе на сертификат из входных параметров. Если такой сертификат не будет найден, данный метод осуществит выпуск нового сертификата по запросу аналогично методу выпуска сертификата по запросу (см. раздел 1.16). Если сертификат с аналогичным указанному в запросе на сертификат «Subject Key Identifier» будет найден, программа: 1) проверит статус найденного сертификата. Если срок действия данного сертификата истек, запрос пользователя будет отклонен.

2) проверит соответствие значений в полях SDN и SAN, указанных в запросе (или во входных параметрах метода), значениям в полях SDN и SAN в найденном сертификате. При соответствии значений будет осуществлен выпуск сертификата по запросу на сертификат из входных параметров метода, иначе запрос пользователя будет отклонен.	
URL – certificate-authority-service/api/v2a/public/certificates/renewal/{caId}/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A сертификаты/renewalRequestByCald_1	
Query	
{	
caId (UUID),	ID ЦС
subjectId (UUID) [обязателен, если не указан userId],	ID субъекта
userId (UUID) [обязателен, если не указан subjectId]	ID учетной записи
}	
Request	
{	
templateId (UUID),	Идентификатор шаблона
request: {	Запрос на сертификат
contentType(string) [опционально],	Тип загружаемого файла (HTTP MediaType) – application/octet-stream)
fileName (string) [опционально],	Имя загружаемого файла
data (string:binary)	Содержимое PEM файла запроса на сертификат (массив байт в Base64) – см. пример использования метода ниже
},	
subjectName: {	Поля отличительного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectName, переопределяют значения соответствующих полей SDN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	
}, [опционально]	
subjectAltName: {	Поля альтернативного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectAltName, переопределяют значения соответствующих полей SAN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	
} [опционально]	
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID сертификата
downloadActions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
fingerprint (string),	Фингерпринт шаблона
serialnumber (string),	Серийный номер сертификата
templateId (UUID),	ID шаблона
templateName (string),	Имя шаблона
name (string),	Имя сертификата (на основе CN)

<pre> issuerDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] }, </pre>	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
<pre> subjectDN: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] }, </pre>	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
<pre> subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] }, </pre>	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant)	Дата окончания действия сертификата (ISO 8601)
}	

1.18 Методы валидации запроса pkcs#10

1.18.1 Метод валидации запроса pkcs#10 (формат запроса на сертификат из входных параметров – multipart/form-data)

POST API – Валидация запроса pkcs#10 (multipart/form-data)	
Метод доступен: – администратору; – оператору при наличии полномочий на управление субъектами и использование шаблона, идентификаторы которых передаются во входных параметрах. Использование данного метода оператором для валидации запросов на сертификат для учетных записей должно быть запрещено.	
URL – certificate-authority-service/api/v2a/public/certificates/validate/{caId}/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A сертификаты/validate_1	
Query	
{	
caId (UUID)	ID ЦС
subjectId (UUID) [обязателен, если не указан accountId],	ID субъекта
accountId (UUID) [обязателен, если не указан subjectId]	ID учетной записи
}	
Request	

{	
templateId (UUID),	Идентификатор шаблона
request (binary),	Файл запроса на сертификат
subjectName: { (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[] }, [опционально]	Поля отличительного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectName, переопределяют значения соответствующих полей SDN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
subjectAltName: { (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[] } [опционально]	Поля альтернативного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectAltName, переопределяют значения соответствующих полей SAN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
name (string),	Имя сертификата (на основе CN)
templateId (UUID),	ID шаблона
templateName (string),	Имя шаблона
subjectId (UUID)	ID субъекта (может отсутствовать, если в Query указан accountId, а не subjectId)
valid (boolean),	Флаг: запрос прошел валидацию
subjectNames: [{	Компоненты имени субъекта сертификата
fieldName (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN),	Имя компонента
fieldValue (string),	Значение компонента
required (boolean),	Флаг: компонент обязательный
additional (boolean),	Флаг: компонент дополнительный
valid (boolean),	Флаг: компонент прошел валидацию
message (string)	Дополнительное сообщение
}],	
subjectAltNames: [{	Компоненты расширенного имени субъекта сертификата
fieldName (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN),	Имя компонента
fieldValue (string),	Значение компонента
required (boolean),	Флаг: компонент обязательный
additional (boolean),	Флаг: компонент дополнительный
valid (boolean),	Флаг: компонент прошел валидацию
message (string)	Дополнительное сообщение
}]	
}	

1.18.2 Метод валидации запроса pkcs#10 (формат запроса на сертификат из входных параметров – application/json)

Примечание: отличается от метода выше форматом передаваемого во входных параметрах запроса на сертификат. В данном методе на вход поступает содержимое файла запроса на сертификат в формате PEM (содержимое в Base64).

POST API – Валидация запроса pkcs#10 (application/json)	
Метод доступен: – администратору; – оператору при наличии полномочий на управление субъектами и использование шаблона, идентификаторы которых передаются во входных параметрах. Использование данного метода оператором для валидации запросов на сертификат для учетных записей должно быть запрещено.	
URL – certificate-authority-service/api/v2a/public/certificates/validate/{caId}/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A сертификаты/validate_1	
Query	
{	
caId (UUID)	ID ЦС
subjectId (UUID) [обязателен, если не указан accountId],	ID субъекта
accountId (UUID) [обязателен, если не указан subjectId]	ID учетной записи
}	
Request	
{	
templateId (UUID),	Идентификатор шаблона
request: {	Файл запроса на сертификат
contentType (string) [опционально],	Тип загружаемого файла (HTTP MediaType) – application/octet-stream)
fileName (string) [опционально],	Имя загружаемого файла
data (string:binary)	Содержимое загружаемого файла (массив байт в Base64)
},	
subjectName: {	Поля отличительного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра.
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Значения полей, указанные в subjectName, переопределяют значения соответствующих полей SDN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
}, [опционально]	
subjectAltName: {	Поля альтернативного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра.
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPD_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Значения полей, указанные в subjectAltName, переопределяют значения соответствующих полей SAN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
} [опционально]	
}	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
name (string),	Имя сертификата (на основе CN)
templateId (UUID),	ID шаблона
templateName (string),	Имя шаблона

subjectId (UUID)	ID субъекта (может отсутствовать, если в Query указан accountId, а не subjectId)
valid (boolean),	Флаг: запрос прошел валидацию
subjectNames: [{	Компоненты имени субъекта сертификата
fieldName (enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN),	Имя компонента
fieldValue (string),	Значение компонента
required (boolean),	Флаг: компонент обязательный
additional (boolean),	Флаг: компонент дополнительный
valid (boolean),	Флаг: компонент прошел валидацию
message (string)	Дополнительное сообщение
}],	
subjectAltNames: [{	Компоненты расширенного имени субъекта сертификата
fieldName (enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN),	Имя компонента
fieldValue (string),	Значение компонента
required (boolean),	Флаг: компонент обязательный
additional (boolean),	Флаг: компонент дополнительный
valid (boolean),	Флаг: компонент прошел валидацию
message (string)	Дополнительное сообщение
}]	
}	

1.19 Метод поиска сертификатов

GET API – Поиск сертификатов	
Метод доступен администратору и оператору	
URL – certificate-authority-service/api/v2a/public/certificates	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a	
Контроллер%3A сертификаты/findAll_14	
Query	
{	
search (string) [опционально],	Полнотекстовый поиск (имя или серийный номер)
issuerId (UUID) [опционально],	ID сертификата издателя
templateId (UUID) [опционально],	ID шаблона
status (enum[]: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN) [опционально],	Статус сертификата
type (enum[]: CERTIFICATE, ROOT_CA, SUB_CA, UNKNOWN) [опционально],	Тип сертификата
revocationReason (enum[]: UNSPECIFIED, KEY_COMPROMISE, CA_COMPROMISE, AFFILIATION_CHANGED, SUPERSEDED, CESSATION_OF_OPERATION, CERTIFICATE_HOLD, REMOVE_FROM_CRL, PRIVILEGE_WITHDRAWN, AA_COMPROMISE, UNKNOWN) [опционально],	Причина отзыва
notRevocationReason (enum[]: UNSPECIFIED, KEY_COMPROMISE, CA_COMPROMISE, AFFILIATION_CHANGED, SUPERSEDED, CESSATION_OF_OPERATION, CERTIFICATE_HOLD, REMOVE_FROM_CRL, PRIVILEGE_WITHDRAWN, AA_COMPROMISE, UNKNOWN) [опционально],	Исключая причину отзыва
revocationDateFrom (instant) [опционально],	Дата отзыва (начало)
revocationDateTo (instant) [опционально],	Дата отзыва (окончание)
hasRevocationReason (boolean) [опционально],	Флаг: наличие причины отзыва

hasRequest (boolean) [опционально],	Флаг: наличие pkcs10
hasCA (boolean) [опционально],	Флаг: наличие ЦС
isManagementCA (boolean) [опционально],	Флаг: технологический ЦС
isValid (boolean) [опционально],	Флаг: сертификат действует
isExpired (boolean) [опционально],	Флаг: сертификат истек
validFromFrom (instant) [опционально],	Дата начала действия (начало)
validFromTo (instant) [опционально],	Дата начала действия (окончание)
validToFrom (instant) [опционально],	Дата окончания действия (начало)
validToTo (instant),	Дата окончания действия (окончание)
updatedFrom (instant) [опционально],	Дата обновления сертификата (начало)
updatedTo (instant),	Дата обновления сертификата (конец)
subjectId (UUID[]),	ID субъекта
userId (UUID[]),	ID учетной записи
serialnumber (string[]),	Серийный номер
fingerprint (string[]),	Отпечаток
subjectKeyIdentifier (string[]),	Идентификатор ключа субъекта
notId (UUID[]),	Исключая ID сертификата
sortDirection (string),	Направления сортировки (ASC;DESC)
sortBy (string[]),	Список полей, к которым применяется сортировка
pageOffset (integer),	Смещение от начала списка (пагинация)
pageLimit (integer),	Ограничение на размер выборки (пагинация)
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	Идентификатор сертификата
issuerId (UUID),	Идентификатор издателя сертификата
issuerFingerprint (string),	Фингерпринт издателя сертификата
serialnumber (string),	Серийный номер сертификата
fingerprint (string),	Фингерпринт сертификата
issuerDN: {	Имя субъекта издателя сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectAltName: {	Альтернативное имя субъекта сертификата
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
name (string),	Имя сертификата (на основе CN)
templateId (UUID),	Идентификатор шаблона
templateName (string),	Имя шаблона
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant),	Дата окончания действия сертификата (ISO 8601)

status (enum: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN),	Статус сертификата ЦС
isValid (boolean),	Флаг: сертификат действует
isExpired (boolean),	Флаг: сертификат истек
actions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
revocation: {	Сведения об отзыве сертификата
date (instant),	Дата отзыва
number (int32),	Код причины отзыва
value (string)	Значение причины отзыва
},	
hashAlgorithm (enum: SHA1, SHA256, SHA384, SHA512, SHA3_256, SHA3_384, SHA3_512, RSASSA_PSS, MD5, GOST_R_34_11_2012, UNKNOWN),	Алгоритм подписи сертификата
keyAlgorithm (enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN),	Алгоритм ключа сертификата
keyBits (int4),	Длина ключа сертификата
subjectKeyIdentifier (string),	Идентификатор ключа сертификата
authorityKeyIdentifier (string),	Идентификатор ключа издателя сертификата
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.19.1 Пример использования метода

```
Curl -k --location 'https://172.17.152.213/certificate-authority-
service/api/v2a/public/certificates' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiaOiIwNjQxN2YzMC00YjY4LTRkYz
AtODAzZS05OGJlN2FkMjU1NDgiLCJpYXQiOiJlMDIzMDY0NzUsImV4cCI6MTcwMjMwNjY1NX0.BdJq0S8lPuJ
dlnnkVZ1z50eCrkoPPelFnOo0lvt4Q0YaZGm-ckRmriMlsXWqfN__5UbiTBu43QoORHiYLSlJi-NfUo-G-
teXFkNDYreguAY86e8w3PVYEqdSONldxxzGTQZbwznq6-
GG2nhbmVHNwe5Bixx_z8UBVdcdcC4JARQY6RY6_8g00OoI0OSvE7Grgspuq8We49DhYCC35rHwtQGVfNThfpe
25lK_Y4kp7azdKFlgqZhWyNpS6CEy_zm-ZgV7wYezvT8B0f_gGAHW24Zfb73WPO-dY-
zfnJdVlRgcb0vS9cVBwvOaDjbKr_2Jl00fyegZ5UDLWCulhtXLg'
```

1.19.2 Пример ответа на обращение к методу

```
{
  "status": 200,
  "data": {
    "range": {
      "count": 562147,
      "offset": 0,
      "limit": 10,
      "totalPages": 56215,
      "pageNumber": 0
    },
    "items": [
      {
        "id": "00001a1f-75cf-4f3d-8160-fb3ac46b937d",
        "issuerId": "6f89cfdb-fcb8-4d8a-a0f8-e258fc1c4b5f",
        "issuerFingerprint": null,
        "serialnumber": "4d6b853f3eed0e5c9029c49ae208393d3c7da846",
        "fingerprint": "1709af7e2719039b5666e9d1cf35cd45f2e61816",
        "issuerDN": {
          "CN": [
```

```
        "SUB_CA_INFORM"
      ],
      "OU": [
        "Department"
      ],
      "O": [
        "organization"
      ],
      "L": [
        "City"
      ],
      "C": [
        "RU"
      ],
      "DC": [
        "Component"
      ]
    },
    "subjectDN": {
      "CN": [
        "UserTest1425"
      ]
    },
    "subjectAltName": {
      "RFC822NAME": [
        "user_test1425@example.com"
      ],
      "MS_UPN": [
        "user_test1425@example.com"
      ]
    },
    "name": "UserTest1425",
    "templateId": "aa03e458-50cd-46b8-82cd-d5612ed3b647",
    "templateName": "Smartcard Logon",
    "validFrom": "2024-02-01T16:04:36Z",
    "validTo": "2026-01-31T16:04:36Z",
    "status": "REVOKE",
    "isValid": true,
    "isExpired": false,
    "actions": {
      "pl2": true,
      "pem": true,
      "csr": false
    },
    "revocation": {
      "date": "2024-02-02T15:31:45Z",
      "number": 2,
      "value": "Компрометация центра сертификации"
    },
    "hashAlgorithm": "SHA256",
    "keyAlgorithm": "RSA",
    "keyBits": 2048,
    "subjectKeyIdentifier": "14f2bde31d8dfc74e7d2483e609e4147bfa17dac",
    "authorityKeyIdentifier": "7c18ee16b61ff0163e230f147a5da9891ba8208d",
    "created": "2024-02-01T16:04:36.328403Z",
```

```

    "updated": "2024-02-02T15:31:45.495858Z"
  }
]
}
}

```

1.20 Метод получения сертификата по идентификатору

GET API – Получение сертификата по идентификатору	
Метод доступен администратору и оператору при наличии полномочий	
URL – certificate-authority-service/api/v2a/public/certificates/{id}	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3А сертификаты/findById_13	
Query	
{	
id (UUID)	ID сертификата
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	Идентификатор сертификата
chain: {	Цепочка сертификатов (рекурсивный объект)
id (UUID),	Идентификатор сертификата
name (string),	Имя сертификата (на основе CN)
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
issuer: {...}	Издатель сертификата (вложенный объект)
},	
serialnumber (string),	Серийный номер сертификата
fingerprint (string),	Фингерпринт сертификата
name (string),	Имя сертификата (на основе CN)
issuerId (UUID),	Идентификатор издателя сертификата
issuerDN: {	Имя субъекта издателя сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectAltName: {	Альтернативное имя субъекта сертификата

(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
keyAlgorithm (enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN),	Алгоритм ключа сертификата ЦС
hashAlgorithm (enum: SHA1, SHA256, SHA384, SHA512, SHA3_256, SHA3_384, SHA3_512, RSASSA_PSS, MD5, GOST_R_34_11_2012, UNKNOWN),	Алгоритм подписи сертификата ЦС
keyBits (int32),	Длина ключа сертификата ЦС
templateId (UUID),	Идентификатор шаблона
templateName (string),	Имя шаблона
type (enum: CERTIFICATE, ROOT_CA, SUB_CA, UNKNOWN),	Тип сертификата
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant),	Дата окончания действия сертификата (ISO 8601)
status (enum: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN),	Статус сертификата
isValid (boolean),	Флаг: сертификат действует
isExpired (boolean),	Флаг: сертификат истек
actions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
publicKey (string),	Открытый ключ
certificateType (string),	Тип сертификата (X.509)
version (int32),	Версия сертификата
subjectKeyIdentifier (string),	Идентификатор ключа сертификата
authorityKeyIdentifier (string)	Идентификатор ключа издателя сертификата
keyUsages: {	Назначение ключа сертификата
id (uuid),	Идентификатор элемента
code (enum: DIGITAL_SIGNATURE, NON_REPUDIATION, KEY_ENCRYPTMENT, DATA_ENCRYPTMENT, KEY_AGREEMENT, KEY_CERT_SIGN, CRL_SIGN, ENCRYPT_ONLY, DECRYPT_ONLY, UNKNOWN),	Перечисление использования ключа
value (string),	Наименование элемента
description (string)	Описание использования ключа
},	
extendedKeyUsages: {	Расширенное назначение ключа сертификата
id (uuid),	Идентификатор назначения
value (string),	Наименование элемента
oid (string),	OID назначения
description (string),	Описание использования ключа
updated (instant),	Время обновления (ISO 8601)
created (instant),	Время создания (ISO 8601)
default (boolean)	Флаг: расширенное использование по умолчанию
},	Описание OID
ca (boolean),	Флаг: сертификат ЦС
revocation: {	Сведения об отзыве сертификата ЦС
date (instant),	Дата отзыва
number (int4),	Код причины отзыва
value (string)	Значение причины отзыва
},	
aiaUrls (string[]),	URL AIA
ocspUrls (string[]),	URL OCSP
crlUrls (string[]),	URL CRL
deltaCrlUrls (string[]),	URL Delta CRL
userId (uuid),	Идентификатор субъекта
subjectId (uuid),	Идентификатор субъекта
created (instant)	Время создания (ISO 8601)
updated (instant),	Время обновления (ISO 8601)

```
} 
```

1.20.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/certificate-authority-  
service/api/v2a/public/certificates/00001a1f-75cf-4f3d-8160-fb3ac46b937d' \  
--  
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiJiZDk2OWJiNC02NmEzLTQ4MD  
ItYjM0YiliZWUwNzAyOGU3ZGUiLCJpYXQiOiJlZDk2OWJiNC02NmEzLTQ4MDItYjM0YiliZWUwNzAyOGU3ZGU  
XloEjl7zGjPD3D_Y7pVEGUjf88R41Rb7fhmliqA1QuQ4S4ruHG1X5SHZRWzxR20dkRqdL2Vw9YlBwsI4jL3By  
vku3cG2kgbtgp8qxghmBiONQGcn71EAha7GqZXpAj-gAwgVcvQM5udqtn6mbNgC-  
tGb5celeeSG8QEhY74pTYvt7T_Ta_-  
BlpyDNIZnYKYZl2bC_4UTwPkAhtSOJLisenGj8qdQteuhKWN2JnYOxomTfJwJDo7pxVuCmh3POAquRzheKK73  
HZEaI4WfIidfvGMq-4izVknlqrNv8X514OAj_XYsSL6QAHSEiUNnhmsaDlrhvwUsV_g'
```

1.20.2 Пример ответа на обращение к методу

```
{  
  "status": 200,  
  "data": {  
    "id": "00001a1f-75cf-4f3d-8160-fb3ac46b937d",  
    "chain": {  
      "id": "00001a1f-75cf-4f3d-8160-fb3ac46b937d",  
      "name": "UserTest1425",  
      "subjectDN": {  
        "CN": [  
          "UserTest1425"  
        ]  
      },  
      "issuer": {  
        "id": "6f89cfdb-fcb8-4d8a-a0f8-e258fc1c4b5f",  
        "name": "SUB_CA_INFORM",  
        "subjectDN": {  
          "CN": [  
            "SUB_CA_INFORM"  
          ],  
          "OU": [  
            "Department"  
          ],  
          "O": [  
            "organization"  
          ],  
          "L": [  
            "City"  
          ],  
          "C": [  
            "RU"  
          ],  
          "DC": [  
            "Component"  
          ]  
        },  
        "issuer": {  
          "id": "7501beab-e119-4e66-b853-6d9d2c9dc311",  
          "name": "aladdin",  

```

```
        "subjectDN": {
            "CN": [
                "aladdin"
            ],
            "OU": [
                "Department"
            ],
            "O": [
                "organization"
            ],
            "L": [
                "City"
            ],
            "C": [
                "RU"
            ],
            "DC": [
                "Component"
            ]
        },
        "issuer": null
    }
}
},
"serialnumber": "4d6b853f3eed0e5c9029c49ae208393d3c7da846",
"fingerprint": "1709af7e2719039b5666e9d1cf35cd45f2e61816",
"name": "UserTest1425",
"issuerId": "6f89cfdb-fcb8-4d8a-a0f8-e258fc1c4b5f",
"issuerDN": {
    "CN": [
        "SUB_CA_INFORM"
    ],
    "OU": [
        "Department"
    ],
    "O": [
        "organization"
    ],
    "L": [
        "City"
    ],
    "C": [
        "RU"
    ],
    "DC": [
        "Component"
    ]
},
"subjectDN": {
    "CN": [
        "UserTest1425"
    ]
},
"subjectAltName": {
    "RFC822NAME": [
```

```

        "user_test1425@example.com"
    ],
    "MS_UPN": [
        "user_test1425@example.com"
    ]
},
"keyAlgorithm": "RSA",
"hashAlgorithm": "SHA256",
"keyBits": 2048,
"templateId": "aa03e458-50cd-46b8-82cd-d5612ed3b647",
"templateName": "Smartcard Logon",
"type": "CERTIFICATE",
"validFrom": "2024-02-01T16:04:36Z",
"validTo": "2026-01-31T16:04:36Z",
"status": "REVOKE",
"isValid": true,
"isExpired": false,
"actions": {
    "p12": true,
    "pem": true,
    "csr": false
},
"publicKey": "Sun RSA public key, 2048 bits\n params: null\n modulus:
1668452414795304620891460325515804559273932771713666663418253667660917359634940956497
6055764239529437203757589454401711720660963381814054609190027425234429644565424993223
7475799999072342706380345968308260271286547998034449053434279835131742191516962331300
7010396265051055167666632897303070400313737446851110130205031064669376827580379072729
7109325921029427451316427896614112688527247678871772419372494170382138289087148488994
6706273399732625564382938846073892376984213038925695630293027415533793155097972559487
4826122162049086545550642717592627671806009102374430097796539995923372869020597530952
1496783948010104967979\n public exponent: 65537",
"certificateType": "X.509",
"version": 3,
"subjectKeyIdentifier": "14f2bde31d8dfc74e7d2483e609e4147bfa17dac",
"authorityKeyIdentifier": "7c18ee16b61ff0163e230f147a5da9891ba8208d",
"keyUsages": [
    {
        "id": "31a1b2a8-e197-47e4-8cd7-2d463f83166f",
        "code": "DIGITAL_SIGNATURE",
        "value": "Digital signature",
        "description": "Цифровая подпись"
    },
    {
        "id": "519e5b6c-9d21-4478-97dd-f0e09c593151",
        "code": "NON_REPUDIATION",
        "value": "Non-repudiation",
        "description": "Подтверждение подлинности"
    },
    {
        "id": "4973c634-29a2-4f6a-903e-e6b2313ab0eb",
        "code": "KEY_ENCIPHERMENT",
        "value": "Key encipherment",
        "description": "Шифрование ключей"
    }
]

```

```

        "id": "834d6f41-5d03-43e2-9b1c-82da036eed75",
        "code": "DATA_ENCIPHERMENT",
        "value": "Data encipherment",
        "description": "Шифрование данных"
    },
],
"extendedKeyUsages": [
    {
        "id": "b170dc22-c502-4c90-800c-d883f2105c44",
        "code": "CLIENT_AUTHENTICATION",
        "value": "Client Authentication",
        "oid": "1.3.6.1.5.5.7.3.2",
        "description": "Аутентификация клиента"
    },
    {
        "id": "0848b618-2a61-42f8-be8d-192b87d73f92",
        "code": "EMAIL_PROTECTION",
        "value": "Email Protection",
        "oid": "1.3.6.1.5.5.7.3.4",
        "description": "Защита электронной почты"
    },
    {
        "id": "9f40092b-d5e1-4c34-8912-45ddc6971f2e",
        "code": "MS_SMART_CARD_LOGON",
        "value": "MS Smart Card Logon",
        "oid": "1.3.6.1.4.1.311.20.2.2",
        "description": "Вход с MS смарт-картой"
    }
],
"ca": false,
"revocation": {
    "date": "2024-02-02T15:31:45Z",
    "number": 2,
    "value": "Компрометация центра сертификации"
},
"aiaUrls": [
    "http://192.168.0.125:8080/aecaCdp/api/v2a/aia/get-aia/2"
],
"ocspUrls": [
    "http://192.168.0.125:8080/aeca-va/ocsp"
],
"crlUrls": [
    "http://192.168.0.125:8080/aecaCdp/api/v2a/crl/get-crl/1"
],
"deltaCrlUrls": [],
"userId": null,
"subjectId": null,
"created": "2024-02-01T16:04:36.328403Z",
"updated": "2024-02-02T15:31:45.495858Z"
}
}

```

1.21 Метод получения сертификата по серийному номеру

GET API – Получение сертификата по его серийному номеру

Метод доступен администратору и оператору при наличии полномочий	
URL – certificate-authority-service/api/v2a/public/certificates/serialNumber/{serialnumber}	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A сертификаты/getBySerialNumber	
Query	
{	
serialnumber (string)	Серийный номер сертификата (формат: 40 символов, нижний регистр)
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	Идентификатор сертификата
chain: {	Цепочка сертификатов (рекурсивный объект)
id (UUID),	Идентификатор сертификата
name (string),	Имя сертификата (на основе CN)
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
issuer: {...}	Издатель сертификата (вложенный объект)
},	
serialnumber (string),	Серийный номер сертификата
fingerprint (string),	Фингерпринт сертификата
name (string),	Имя сертификата (на основе CN)
issuerId (UUID),	Идентификатор издателя сертификата
issuerDN: {	Имя субъекта издателя сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
subjectAltName: {	Альтернативное имя субъекта сертификата
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
keyAlgorithm (enum: RSA, ECDSA, GOST R 34 10 2012, UNKNOWN),	Алгоритм ключа сертификата ЦС

hashAlgorithm (enum: SHA1, SHA256, SHA384, SHA512, SHA3_256, SHA3_384, SHA3_512, RSASSA_PSS, MD5, GOST_R_34_11_2012, UNKNOWN),	Алгоритм подписи сертификата ЦС
keyBits (int32),	Длина ключа сертификата ЦС
templateId (UUID),	Идентификатор шаблона
templateName (string),	Имя шаблона
type (enum: CERTIFICATE, ROOT_CA, SUB_CA, UNKNOWN),	Тип сертификата
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant),	Дата окончания действия сертификата (ISO 8601)
status (enum: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN),	Статус сертификата
isValid (boolean),	Флаг: сертификат действует
isExpired (boolean),	Флаг: сертификат истек
actions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
publicKey (string),	Открытый ключ
certificateType (string),	Тип сертификата (X.509)
version (int32),	Версия сертификата
subjectKeyIdentifier (string),	Идентификатор ключа сертификата
authorityKeyIdentifier (string)	Идентификатор ключа издателя сертификата
keyUsages: {	Назначение ключа сертификата
id (uuid),	Идентификатор элемента
code (enum: DIGITAL_SIGNATURE, NON_REPUDIATION, KEY_ENCRYPTMENT, DATA_ENCRYPTMENT, KEY_AGREEMENT, KEY_CERT_SIGN, CRL_SIGN, ENCRYPT_ONLY, DECRYPT_ONLY, UNKNOWN),	Перечисление использования ключа
value (string),	Наименование элемента
description (string)	Описание использования ключа
},	
extendedKeyUsages: {	Расширенное назначение ключа сертификата
id (uuid),	Идентификатор элемента
value (string),	Наименование элемента
oid (string),	OID назначения
description (string),	Описание использования ключа
updated (instant),	Время обновления (ISO 8601)
created (instant),	Время создания (ISO 8601)
default (boolean)	Флаг: расширенное использование по умолчанию
},	Описание OID
ca (boolean),	Флаг: сертификат ЦС
revocation: {	Сведения об отзыве сертификата ЦС
date (instant),	Дата отзыва
number (int4),	Код причины отзыва
value (string)	Значение причины отзыва
},	
aiaUrls (string[]),	URL AIA
ocspUrls (string[]),	URL OCSP
crlUrls (string[]),	URL CRL
deltaCrlUrls (string[]),	URL Delta CRL
userId (uuid),	Идентификатор субъекта
subjectId (uuid),	Идентификатор субъекта
created (instant)	Время создания (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
}	

1.21.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/certificate-authority-
service/api/v2a/public/certificates/serialNumber/4d6b853f3eed0e5c9029c49ae208393d3c7d
a846' \
```

```
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiI1MjQ4OWFmYS0zM2MzLTJTRiNz
YtOTk2ZS0yOWZiNzBkMTIiLCJpYXQiOiJlMjQ4OWFmYS0zM2MzLTJTRiNzMDIzMDg5MTIsImV4cCI6MTcwMjMwOTA5Mn0.SAGyoqmqiTY
rGwTP-
HtYJS705ScACS0c7QojvB7no1EZfKnhhLosVyJXoe_XgttVyD0Jk_OPszQqsH0P2xMzAcrrs1pmqtGjyW_ei
vnX4DhGaQZDrHldljTXTa9_iI0WfLt5-oVK8VKN3tUaLwAOZNcgbaYDP1T018Wkb4kx5euj-
8z9xrBP8MWSXjybIt4CD9WkmpF9qF1liu6DaTHkiDvFgpCrNF0lC6bmbAWNag8eo95M1JTQpOILFiuA2vzVgL
YW9TdCw6KZ8e0059OTgcZFuMqjKLaiqdZm3vK9o_fmiAvjvMVRBQQy8aP4IcCmHf7x40mo5D5bg7Z3IOvGA'
```

1.21.2 Пример ответа на обращение к методу

```
{
  "status": 200,
  "data": {
    "id": "00001a1f-75cf-4f3d-8160-fb3ac46b937d",
    "chain": {
      "id": "00001a1f-75cf-4f3d-8160-fb3ac46b937d",
      "name": "UserTest1425",
      "subjectDN": {
        "CN": [
          "UserTest1425"
        ]
      },
      "issuer": {
        "id": "6f89cfdb-fcb8-4d8a-a0f8-e258fc1c4b5f",
        "name": "SUB_CA_INFORM",
        "subjectDN": {
          "CN": [
            "SUB_CA_INFORM"
          ],
          "OU": [
            "Department"
          ],
          "O": [
            "organization"
          ],
          "L": [
            "City"
          ],
          "C": [
            "RU"
          ],
          "DC": [
            "Component"
          ]
        },
        "issuer": {
          "id": "7501beab-e119-4e66-b853-6d9d2c9dc311",
          "name": "aladdin",
          "subjectDN": {
            "CN": [
              "aladdin"
            ],
            "OU": [
              "Department"
            ]
          }
        }
      }
    }
  }
}
```

```
    ],
    "O": [
        "organization"
    ],
    "L": [
        "City"
    ],
    "C": [
        "RU"
    ],
    "DC": [
        "Component"
    ]
  },
  "issuer": null
}

},
"serialnumber": "4d6b853f3eed0e5c9029c49ae208393d3c7da846",
"fingerprint": "1709af7e2719039b5666e9d1cf35cd45f2e61816",
"name": "UserTest1425",
"issuerId": "6f89cfdb-fcb8-4d8a-a0f8-e258fc1c4b5f",
"issuerDN": {
  "CN": [
    "SUB_CA_INFORM"
  ],
  "OU": [
    "Department"
  ],
  "O": [
    "organization"
  ],
  "L": [
    "City"
  ],
  "C": [
    "RU"
  ],
  "DC": [
    "Component"
  ]
},
"subjectDN": {
  "CN": [
    "UserTest1425"
  ]
},
"subjectAltName": {
  "RFC822NAME": [
    "user_test1425@example.com"
  ],
  "MS_UPN": [
    "user_test1425@example.com"
  ]
},
},
```

```

    "keyAlgorithm": "RSA",
    "hashAlgorithm": "SHA256",
    "keyBits": 2048,
    "templateId": "aa03e458-50cd-46b8-82cd-d5612ed3b647",
    "templateName": "Smartcard Logon",
    "type": "CERTIFICATE",
    "validFrom": "2024-02-01T16:04:36Z",
    "validTo": "2026-01-31T16:04:36Z",
    "status": "REVOKE",
    "isValid": true,
    "isExpired": false,
    "actions": {
      "p12": true,
      "pem": true,
      "csr": false
    },
    "publicKey": "Sun RSA public key, 2048 bits\n params: null\n modulus:
1668452414795304620891460325515804559273932771713666663418253667660917359634940956497
6055764239529437203757589454401711720660963381814054609190027425234429644565424993223
7475799999072342706380345968308260271286547998034449053434279835131742191516962331300
7010396265051055167666632897303070400313737446851110130205031064669376827580379072729
7109325921029427451316427896614112688527247678871772419372494170382138289087148488994
6706273399732625564382938846073892376984213038925695630293027415533793155097972559487
4826122162049086545550642717592627671806009102374430097796539995923372869020597530952
1496783948010104967979\n public exponent: 65537",
    "certificateType": "X.509",
    "version": 3,
    "subjectKeyIdentifier": "14f2bde31d8dfc74e7d2483e609e4147bfa17dac",
    "authorityKeyIdentifier": "7c18ee16b61ff0163e230f147a5da9891ba8208d",
    "keyUsages": [
      {
        "id": "31a1b2a8-e197-47e4-8cd7-2d463f83166f",
        "code": "DIGITAL_SIGNATURE",
        "value": "Digital signature",
        "description": "Цифровая подпись"
      },
      {
        "id": "519e5b6c-9d21-4478-97dd-f0e09c593151",
        "code": "NON_REPUDIATION",
        "value": "Non-repudiation",
        "description": "Подтверждение подлинности"
      },
      {
        "id": "4973c634-29a2-4f6a-903e-e6b2313ab0eb",
        "code": "KEY_ENCIPHERMENT",
        "value": "Key encipherment",
        "description": "Шифрование ключей"
      },
      {
        "id": "834d6f41-5d03-43e2-9b1c-82da036eed75",
        "code": "DATA_ENCIPHERMENT",
        "value": "Data encipherment",
        "description": "Шифрование данных"
      }
    ]
  },

```

```

    "extendedKeyUsages": [
      {
        "id": "b170dc22-c502-4c90-800c-d883f2105c44",
        "code": "CLIENT_AUTHENTICATION",
        "value": "Client Authentication",
        "oid": "1.3.6.1.5.5.7.3.2",
        "description": "Аутентификация клиента"
      },
      {
        "id": "0848b618-2a61-42f8-be8d-192b87d73f92",
        "code": "EMAIL_PROTECTION",
        "value": "Email Protection",
        "oid": "1.3.6.1.5.5.7.3.4",
        "description": "Защита электронной почты"
      },
      {
        "id": "9f40092b-d5e1-4c34-8912-45ddc6971f2e",
        "code": "MS_SMART_CARD_LOGON",
        "value": "MS Smart Card Logon",
        "oid": "1.3.6.1.4.1.311.20.2.2",
        "description": "Вход с MS смарт-картой"
      }
    ],
    "ca": false,
    "revocation": {
      "date": "2024-02-02T15:31:45Z",
      "number": 2,
      "value": "Компрометация центра сертификации"
    },
    "aiaUrls": [
      "http://192.168.0.125:8080/aecaCdp/api/v2a/aia/get-aia/2"
    ],
    "ocspUrls": [
      "http://192.168.0.125:8080/aeca-va/ocsp"
    ],
    "crlUrls": [
      "http://192.168.0.125:8080/aecaCdp/api/v2a/crl/get-crl/1"
    ],
    "deltaCrlUrls": [],
    "userId": null,
    "subjectId": null,
    "created": "2024-02-01T16:04:36.328403Z",
    "updated": "2024-02-02T15:31:45.495858Z"
  }
}

```

1.22 Метод получения сертификата по его отпечатку

GET API – Получение сертификата по его отпечатку	
Метод доступен администратору и оператору при наличии полномочий	
URL – <code>certificate-authority-service/api/v2a/public/certificates/fingerprint/{fingerprint}</code>	
Swagger: <code>https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3А сертификаты/getByFingerprint_1</code>	
Query	

{	
fingerprint (String)	Отпечаток сертификата
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	Идентификатор сертификата
chain: {	Цепочка сертификатов (рекурсивный объект)
id (UUID),	Идентификатор сертификата
name (string),	Имя сертификата (на основе CN)
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
issuer: {...}	Издатель сертификата (вложенный объект)
},	
serialnumber (string),	Серийный номер сертификата
fingerprint (string),	Фингерпринт сертификата
name (string),	Имя сертификата (на основе CN)
issuerId (UUID),	Идентификатор издателя сертификата
issuerDN: {	Имя субъекта издателя сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
subjectAltName: {	Альтернативное имя субъекта сертификата
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
keyAlgorithm (enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN),	Алгоритм ключа сертификата ЦС
hashAlgorithm (enum: SHA1, SHA256, SHA384, SHA512, SHA3_256, SHA3_384, SHA3_512, RSASSA_PSS, MD5, GOST_R_34_11_2012, UNKNOWN),	Алгоритм подписи сертификата ЦС
keyBits (int32),	Длина ключа сертификата ЦС
templateId (UUID),	Идентификатор шаблона
templateName (string),	Имя шаблона
type (enum: CERTIFICATE, ROOT_CA, SUB_CA, UNKNOWN),	Тип сертификата

validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant),	Дата окончания действия сертификата (ISO 8601)
status (enum: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN),	Статус сертификата
isValid (boolean),	Флаг: сертификат действует
isExpired (boolean),	Флаг: сертификат истек
actions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
publicKey (string),	Открытый ключ
certificateType (string),	Тип сертификата (X.509)
version (int32),	Версия сертификата
subjectKeyIdentifier (string),	Идентификатор ключа сертификата
authorityKeyIdentifier (string)	Идентификатор ключа издателя сертификата
keyUsages: {	Назначение ключа сертификата
id (uuid),	Идентификатор элемента
code (enum: DIGITAL_SIGNATURE, NON_REPUDIATION, KEY_ENCRYPTMENT, DATA_ENCRYPTMENT, KEY_AGREEMENT, KEY_CERT_SIGN, CRL_SIGN, ENCRYPT_ONLY, DECRYPT_ONLY, UNKNOWN),	Перечисление использования ключа
value (string),	Наименование элемента
description (string)	Описание использования ключа
},	
extendedKeyUsages: {	Расширенное назначение ключа сертификата
id (uuid),	Идентификатор элемента
value (string),	Наименование элемента
oid (string),	OID назначения
description (string),	Описание использования ключа
updated (instant),	Время обновления (ISO 8601)
created (instant),	Время создания (ISO 8601)
default (boolean)	Флаг: расширенное использование по умолчанию
},	Описание OID
ca (boolean),	Флаг: сертификат ЦС
revocation: {	Сведения об отзыве сертификата ЦС
date (instant),	Дата отзыва
number (int4),	Код причины отзыва
value (string)	Значение причины отзыва
},	
aiaUrls (string[]),	URL AIA
ocspUrls (string[]),	URL OCSP
crlUrls (string[]),	URL CRL
deltaCrlUrls (string[]),	URL Delta CRL
userId (uuid),	Идентификатор субъекта
subjectId (uuid),	Идентификатор субъекта
created (instant)	Время создания (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
}	

1.22.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/certificate-authority-
service/api/v2a/public/certificates/fingerprint/1709af7e2719039b5666e9d1cf35cd45f2e61
816' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiaOiI2ZTA5MDg3OC1kN2QzLTRkZT
gtOTdhMiliYTZyZTA0ODZyZnJYiLCJpYXQiOiJlZ3MDIzMDc2NTUsImV4cCI6MTcwMjMwNzgzNX0.
IBWp4OqcR2T1yJiTeOyzHjMm3m2nFR1CXUb64yfxfe0rZeJ4BnTILcqYsQdFCUNcWk8e9xVnoLEGGdx6cDNyXyBrJUS2Idc
vzMtns7m9EQCqtrvtispInoLtGqYsuh_ZlxsIQxm3uTX-cfTedLgdrjgrgJCELvyV-w7Rr9-
mjSOhGXoeYqhsCYciwVn29Ye6o-CjB9PIr4HN72W4Ng1yDT0OVSYA_Uld-
```

```
1bu30O87a6L9NoOJlK9wiCKVYv9hDSEfMongWe-  
0MOEBSKsldN9wiRgjQ9Hnk2B5bXzqZM2s8ZeUJ_EHgNx1x5sjwV2iXb3y3tcIRony5EvwYf8X7qPw'
```

1.22.2 Пример ответа на обращение к методу

```
{  
  "status": 200,  
  "data": {  
    "id": "00001a1f-75cf-4f3d-8160-fb3ac46b937d",  
    "chain": {  
      "id": "00001a1f-75cf-4f3d-8160-fb3ac46b937d",  
      "name": "UserTest1425",  
      "subjectDN": {  
        "CN": [  
          "UserTest1425"  
        ]  
      },  
      "issuer": {  
        "id": "6f89cfdb-fcb8-4d8a-a0f8-e258fc1c4b5f",  
        "name": "SUB_CA_INFORM",  
        "subjectDN": {  
          "CN": [  
            "SUB_CA_INFORM"  
          ],  
          "OU": [  
            "Department"  
          ],  
          "O": [  
            "organization"  
          ],  
          "L": [  
            "City"  
          ],  
          "C": [  
            "RU"  
          ],  
          "DC": [  
            "Component"  
          ]  
        },  
        "issuer": {  
          "id": "7501beab-e119-4e66-b853-6d9d2c9dc311",  
          "name": "aladdin",  
          "subjectDN": {  
            "CN": [  
              "aladdin"  
            ],  
            "OU": [  
              "Department"  
            ],  
            "O": [  
              "organization"  
            ],  
            "L": [  
              "City"  
            ]  
          }  
        }  
      }  
    }  
  }  
}
```

```
        ],
        "C": [
            "RU"
        ],
        "DC": [
            "Component"
        ]
    },
    "issuer": null
}

},
"serialnumber": "4d6b853f3eed0e5c9029c49ae208393d3c7da846",
"fingerprint": "1709af7e2719039b5666e9d1cf35cd45f2e61816",
"name": "UserTest1425",
"issuerId": "6f89cfdb-fcb8-4d8a-a0f8-e258fc1c4b5f",
"issuerDN": {
    "CN": [
        "SUB_CA_INFORM"
    ],
    "OU": [
        "Department"
    ],
    "O": [
        "organization"
    ],
    "L": [
        "City"
    ],
    "C": [
        "RU"
    ],
    "DC": [
        "Component"
    ]
},
"subjectDN": {
    "CN": [
        "UserTest1425"
    ]
},
"subjectAltName": {
    "RFC822NAME": [
        "user_test1425@example.com"
    ],
    "MS_UPN": [
        "user_test1425@example.com"
    ]
},
"keyAlgorithm": "RSA",
"hashAlgorithm": "SHA256",
"keyBits": 2048,
"templateId": "aa03e458-50cd-46b8-82cd-d5612ed3b647",
"templateName": "Smartcard Logon",
"type": "CERTIFICATE",
```

```

    "validFrom": "2024-02-01T16:04:36Z",
    "validTo": "2026-01-31T16:04:36Z",
    "status": "REVOKE",
    "isValid": true,
    "isExpired": false,
    "actions": {
      "p12": true,
      "pem": true,
      "csr": false
    },
    "publicKey": "Sun RSA public key, 2048 bits\n  params: null\n  modulus:
1668452414795304620891460325515804559273932771713666663418253667660917359634940956497
6055764239529437203757589454401711720660963381814054609190027425234429644565424993223
7475799999072342706380345968308260271286547998034449053434279835131742191516962331300
7010396265051055167666632897303070400313737446851110130205031064669376827580379072729
7109325921029427451316427896614112688527247678871772419372494170382138289087148488994
6706273399732625564382938846073892376984213038925695630293027415533793155097972559487
4826122162049086545550642717592627671806009102374430097796539995923372869020597530952
1496783948010104967979\n  public exponent: 65537",
    "certificateType": "X.509",
    "version": 3,
    "subjectKeyIdentifier": "14f2bde31d8dfc74e7d2483e609e4147bfa17dac",
    "authorityKeyIdentifier": "7c18ee16b61ff0163e230f147a5da9891ba8208d",
    "keyUsages": [
      {
        "id": "31a1b2a8-e197-47e4-8cd7-2d463f83166f",
        "code": "DIGITAL_SIGNATURE",
        "value": "Digital signature",
        "description": "Цифровая подпись"
      },
      {
        "id": "519e5b6c-9d21-4478-97dd-f0e09c593151",
        "code": "NON_REPUDIATION",
        "value": "Non-repudiation",
        "description": "Подтверждение подлинности"
      },
      {
        "id": "4973c634-29a2-4f6a-903e-e6b2313ab0eb",
        "code": "KEY_ENCIPHERMENT",
        "value": "Key encipherment",
        "description": "Шифрование ключей"
      },
      {
        "id": "834d6f41-5d03-43e2-9b1c-82da036eed75",
        "code": "DATA_ENCIPHERMENT",
        "value": "Data encipherment",
        "description": "Шифрование данных"
      }
    ],
    "extendedKeyUsages": [
      {
        "id": "b170dc22-c502-4c90-800c-d883f2105c44",
        "code": "CLIENT_AUTHENTICATION",
        "value": "Client Authentication",
        "oid": "1.3.6.1.5.5.7.3.2",

```

```

        "description": "Аутентификация клиента"
    },
    {
        "id": "0848b618-2a61-42f8-be8d-192b87d73f92",
        "code": "EMAIL_PROTECTION",
        "value": "Email Protection",
        "oid": "1.3.6.1.5.5.7.3.4",
        "description": "Защита электронной почты"
    },
    {
        "id": "9f40092b-d5e1-4c34-8912-45ddc6971f2e",
        "code": "MS_SMART_CARD_LOGON",
        "value": "MS Smart Card Logon",
        "oid": "1.3.6.1.4.1.311.20.2.2",
        "description": "Вход с MS смарт-картой"
    }
],
"ca": false,
"revocation": {
    "date": "2024-02-02T15:31:45Z",
    "number": 2,
    "value": "Компрометация центра сертификации"
},
"aiaUrls": [
    "http://192.168.0.125:8080/aecaCdp/api/v2a/aia/get-aia/2"
],
"ocspUrls": [
    "http://192.168.0.125:8080/aeca-va/ocsp"
],
"crlUrls": [
    "http://192.168.0.125:8080/aecaCdp/api/v2a/crl/get-crl/1"
],
"deltaCrlUrls": [],
"userId": null,
"subjectId": null,
"created": "2024-02-01T16:04:36.328403Z",
"updated": "2024-02-02T15:31:45.495858Z"
}
}

```

1.23 Метод отзыва (приостановки) сертификата по идентификатору

PUT API – Отзыв (приостановка) сертификата по идентификатору	
Метод доступен администратору и оператору при наличии полномочий	
URL – certificate-authority-service/api/v2a/public/certificates/{id}/revoke	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A%20сертификаты/revokeById	
Query	
{	
id (UUID)	ID сертификата
}	
Request	
{	
reason (enum: UNSPECIFIED, KEY_COMPROMISE, CA_COMPROMISE, AFFILIATION_CHANGED,	Причина отзыва (приостановки)

SUPERSEDED, CESSATION_OF_OPERATION, CERTIFICATE_HOLD, REMOVE_FROM_CRL, PRIVILEGE_WITHDRAWN, AA_COMPROMISE, UNKNOWN)	
}	
Response	
-	
При указании значения «CERTIFICATE_HOLD» в параметре reason сертификат будет приостановлен, а не отозван.	

1.24 Метод активации сертификата по идентификатору

PUT API – Активация сертификата по идентификатору	
Метод доступен администратору и оператору при наличии полномочий	
URL – certificate-authority-service/api/v2a/public/certificates/{id}/reactivate	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A сертификаты/reactivateById	
Query	
{	
id (UUID)	ID сертификата
}	
Request	
-	
Response	
-	

1.25 Метод публикации сертификата в РС по идентификатору

PUT API – Публикация сертификата в ресурсную систему	
Метод доступен администратору и оператору при наличии полномочий	
URL – certificate-authority-service/api/v2a/public/certificates/{id}/publish	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A сертификаты/publishById	
Query	
{	
id (uuid)	Идентификатор сертификата
}	
Request	
-	
Response	
-	

1.26 Метод расшифровки контейнера сертификата

POST API – Расшифровка контейнера сертификата	
Метод доступен администратору	
URL – certificate-authority-service/api/v2a/public/parse/pkcs12	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A контейнеры сертификатов/parse	
Query	
Request	
{	
container: {	Файл контейнера

contentType (string) [опционально],	Тип загружаемого файла (HTTP MediaType) - application/octet-stream)
fileName (string) [опционально],	Имя загружаемого файла
data (string:binary)	Содержимое загружаемого файла (массив байт в Base64)
},	
password(string),	Пароль от контейнера
templateId (UUID)	Идентификатор шаблона
}	
Response	
{	Ответ JSON в HTTP-body
serialnumber (string),	Серийный номер сертификата
fingerprint (string),	Фингерпринт сертификата
name (string),	Имя сертификата (на основе CN)
issuerDN: {	Имя субъекта издателя сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
subjectAltName: {	Альтернативное имя субъекта сертификата
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant),	Дата окончания действия сертификата (ISO 8601)
privateKey: {	Файл закрытого ключа
contentType (string),	Тип загружаемого файла (HTTP MediaType)
fileName (string),	Имя загружаемого файла
data (string:binary)	Содержимое загружаемого файла (массив байт в Base64)
},	
certificate: {	Файл сертификата
contentType (string),	Тип загружаемого файла (HTTP MediaType)
fileName (string),	Имя загружаемого файла
data (string:binary)	Содержимое загружаемого файла (массив байт в Base64)
},	
keyUsages: {	Назначение ключа сертификата
id (uuid),	Идентификатор элемента
code (enum: DIGITAL_SIGNATURE, NON_REPUDIATION, KEY_ENCRYPTMENT, DATA_ENCRYPTMENT, KEY_AGREEMENT, KEY_CERT_SIGN, CRL_SIGN,	Перечисление использования ключа

1.26.1 Пример использования метода

Руководство администратора. Часть 2. Приложение 3. Описание методов REST API
Стр. 94 / 177

Sb/9rRVDrgAIVVxOJxu7aHYhhrSMQ68MwHMOW+AG/Yd6XtJZVS19flTAqQX8ZE44d9BN94V9nNaXbGggmvOJf
cuEF6qDCfikTGvEjPnfJ50slyxmfRcEXonQxpWn4SD6V+f9u/XQBxeQ1rl2us80kiMV0H/GxTI+2ZpChLWZP
KSR/ZKH6b3BaW9bmIf6JgckbgqkXopkTzFEMB8GCSqGSIB3DQEJFDESHhAAbwBwAGUAcgBhAHQAbwByMCEGCS
qGSIB3DQEJFTEUBBJUaW1lIDE3MzgwNjY2NzM4MjkwggxxBgkqhkiG9w0BBwagggxiMIIMXgIBADCCDFcGCSq
GSIB3DQEhATBmBgkqhkiG9w0BBQ0wWTA4BgkqhkiG9w0BBQwwKwQUU1jFCBHSipoBNMsch4ZSn4ans+gCAicQ
AgEgMAwGCCqGSIb3DQIJBQAwHQYJYIZIAWUDBAEqBBByQJ1BAkI6TOS02h57HBHTgIIL4MWWxmn5R/dzwC3Q1
a69vlskZMbJFbVvkVoBNnZviR6RvzE4/heF7BBpJvH7PaHgP8zFpAQqOsYcaWK7Pzx1vOHvYywZRY+4kDC0o7
LMJ/Vb7gQPMjUr9MmdcCJ+b+Gi44UK7ABUDtVy+NDvbTYKexgtZYk22LczatvioBoN46/KzZKNd9N/F2vMvOR
dsIhDKOUOeKXnqlWS1ZpTkgfpcgtvxAwBT2VlOxrVeJ4yG/i2LEHS6GU04xky1lZG82Z/vowYRv/1LxE9oqsI
mkAPX+5FZR4C/sLUYz7hXj3dss3hogmfpeJ+nEvJAnZQFXzbzxiEDdrPWEiA7ghVnpoq++9Y2b+WOFfchWuiy
8REceOpDYkJ5oiwQIGiwSUq9eumDi3sr4JnatT0BSAylKt1CC4vYGuWP8oUrrEPpj+6451AM550B7BebiSqP
tiDeK61crVCS+3q+4b4Q7Upe7AW0o9zMx/F5aCypqHptcJSaAGE8PghI3NHX7xYBfCR/Bwlw63qrFrBjxOJ2y
W3OXyyrX4VyLvNbj28hax1OrhLdMhEX36DP5gWvJVqSIJVIteG4LpssgYCHRZIDGjPgDWA1zlxnTViLMG8BiW
jPaI0GeAsGjbdC28YPKVia7qztKSGQW5uLymkrylbnfy7EGoxLSHDB5dV1rJnA6t1QNI7Gy2EZiPISbEAqaNp
V5kkKOvbjt6VCQ1e/WwGh/fzyRjuwCOFOgWqt0eZbv40nq1ObD0upxQXjOBXBadMPd/Pgf2nKj/131ENJAN5
9Lc0lsJV4Z3fHWbD+My7+Uz0stOhXnjfN/G07kvnlfhyds/8qaNMCLBwXoOtU9i7FGtCqCjpk3TJepS+RASH8
+VhRkTXPnL1IxoTNECn/agS8podOwBFmkqkRvxTPNsaLg7rdxj6UjZvIB4P2cnyRRfMJ0JYT5KvU0C5YLEQpS
OlUUs23OlCAKncFIz87dvjwJ/roRLNg9RJtW5XG6tM2AvV5X2bJu6Q7KvsUfOvBM5ib6z3b4lwZjZ5JPzzhTu
KRi7GtamVJWxcbMZsVPzjBTuLklTEEvdyd+Ru6xOe2rl+QkdxxEiPbjJQDefz0rzBbrrcgNYhwBYqVkhMmW2ZG
w1lZEpfmJQ7pSnZJUR3ziOdt/eXcZ0dtvKpnR+DvEFowAKroGLqSoUoilKZ8fEqlXi7dVS+YkbnvqdD9H2N1V
qLBddtcg4ZkxNayOF0GbGq0cKTduvWdpN2pPEeyndqZVpYoQOKiYtAA+VbqGW2XZwyz+hsg9PzdU5rd8ZVUfp
WTPsKgZ3AclKveYkFkx9ynsQc7cEdRr75kiF7FQURzL0vQXamuKHhuf5OuQJByf5y+BKHFk5GnHUN5eq46ptb
hoWXaPzLpXgCzYribTu/NkA6NfbSyXqtSMAoQznxhagnFa8fP6EVhFYEAckBwe3yy/iirxGWh851MeWe0xyZ
url/cq9vg9zsVhPcpvxY21m2j/51qz0Er1qTXLObJfy7AbSsOmc8CXtb5smss+aB6/bGmj0jLao87WARWlWZU
ye5p3g5SbqWlyARHHTGt/qG5o3Zlw7hblRPYO+q1dZaM3SaHtGvApb3SyNT38WgdD6Jl02G/uCHVCQBftLmki
bbzP8khxr9jJfaSFUGn79X8HFL7w//BRmJ2I5fbtei0biCM1RNMNQ0cfmtapplEccLhzky4fb+MAKjNIUlwE
2SOA52R16gQcvUdUKJ7XcM1HYappd6cjSKdERWjD1FY0ezQSyQbO8+Tf2CNfV3nhOeIvX4PtYKuNwTfe8iJj
yXO9gcTwPGy3tshHVJL+pQkhFBw27y4lfcu5OoWCHzdNkr+CRZ8d25DcZrRiXcBEka0CG5LtSxMafyJeFsK+V
CIh1R4hroWAp2XCdANusE8NB4GJ26Cce6ttCadMoss2KBiC+Zf1ftEEwks9h1EoHvhaIIfeY2h1HsgB8R4lhO
gfSPgCdJu+XBseRRetohQObtVWOQUt472f8MUpJcA67Poljub6dYpAdJ1AVlnsw7YLCnwe0T4M6dj9v8ZnNBc
kvX8COO2hR5NuZqqhmlrrOIyRDdzx8Rz8k+DYheYHu+33hyWVflj79fO2AnmDVH8KCOOWslCqOoYQDCy2vyDb
ZeBfcv12AIBAOVfzsjqMMjnaue2ErJeZLaqdg+58s6BiQE2XaWA1EM7cpaWr/frSsBeGYe2ghpKiqonn1IJwv
Wfzgl7dekLevfGmU5/GqCFwhF3chrBN2TANGMJ2gTwboMElu7C2Z2Gm5KqrGwlf+HW4YGdvbOtM8ff4Fi1hJw
E+MhEqsn4A7c0wbK9bGMC2Q8s8QXHfoQjEbpG9BW/VBouDgWY4IbKRRv9USv2M2juiWSF+2v8K1kKD6gvCO
f8dKerXjBKMdxZzo5MyAKmOwGMFECAFYHrEoOqQ+ZTK7YtJUAUM2zeFJ/iq65YbVqYVh04m+dIwMBnns6lLcb
NfiH/CUZ7US9pBKnuLd3K7mj2HcJ41Nw999wgrW+6q3wdppSyg3M88woOhd79fvLiCbYVsy2O9wH+78aXziMa
klTldBOBH56RwBtVa034bCBC+srWjs3WAU5qrFxUGjEsPtxpELCyNSQRco3L32W6J4YNYrNUQtSrOfk0NZqPr
gsoLquN/bce41LFqlGPZuOdGwbD86y+989VcsKWImqMykFkbwul4APNwTJOqjLPo03o/plKKs+4Ejj5BCKvVm
c+IPVTr2oHh+n1hWsPaq/vSlmAr6X1TKkID33BLH4dLZpj5WtYrNLYC2fqnmMrdlPY2eg7eJeeSfWlMu21AXL
iWSHK16GQ2sWs3DxveK6Eg4dgEb17M7Uf5Ru1G5Gp1lwSHyftstPjTG7EKQ7pyZkxH9FnbB7AhlNhSIvte8Sb
fKfgiSfGVGDCodlrKHnM5ioPjOGbRfr5f25nklz/BXm59Waw24C4aadq1ggAJZbFugXFvVOVCHD704YZqUj0/
yioB6wJpQ0clzMYQnl/7+/nbwtV/tBdR2tti/Yym54V0UfNXQS1oKvSiteLypTMLaZyi3w0LT0fvOb10MQGX3
LBaCt4aZtz01Q+6gGD10ZetDPz3IbYj9vEwaBTTip8SrL053h93EZixSpXIDnwXw07BvbAhDWOH1+CxhS0Sm4
zOLykvwynZFBGdz0Z6/sp66CSb5GaXPXl+30eaZa80K26PsXeEhAFYUy/c2O3AD7V3vukLiDpgxPB4kmfd1wW
JsRCjjiiyVGJnqnCjDeSrEblUm9RTibeVdlvMEf4cf1rKiefIbxVWbw9MeMSR31ZuTiaqyG35mVwCtfJuJI/9
NQ4SsfARQGettEuPsVfoRS7Kf/gNrMrwZ41PiZsvEEW16WO+NiE+lu2b7e33lgIlVukSyppgDQWzY583gtYV3
xxnVG7y4/PDPPPO/LIX1lQ2I4U/jDN3kjlDaHPuMPwQhulSMta8fXajW05F/JPiExY4xAVPHNaGLrsaCULZm+
rpARwwDsYD7vXWMqPwWXYDN57sTTLdYS9KH9MxcBqkCoF7uTNiamT7Yn1Mo1ZF1YJ9tS6xjnu3IJhGLK7mTx5
6fPoD50SrrC+diMF4bccd48yJ9J+xOVU817ScdLftl4xynaLpAQOYR3IIcu3fwhZs1d80EJ204DV3gT+E34io
dX8An6OYG/pkgjMtZdVfojwMtaX0Iu6r/aBd0bgUst7nj6Ew+IApwjQ/p+zKrJbHH4wfWiVLZXloRVtZJ9F88
936uz/Ad5VNSHGLrL7pGit65x1PQmjQwVaj5g9bD+FVbffd7597q3+aKpO2oFWtJuhajJqVXkAKj0RLlLfCdZ+
oLXeELzVJjydwQ1h3vbZ22KCJLEb+pM6vo37gr9fHMkwP+sSlNuivARXZsUaLBLtgTx/ZRlca9VokDyMeroj6
XrqXyNqYIfxzYHQ8KWwulFd68GVLMeFGplHIjeZB4vuSLOOusjI2mkudfI9ICc9nBcCf2Mmi/3zbPAYsnxV8p
8SF3cbQPnSltcmGmhgVflUiWBsCBLsrNoM4u4QPW49YwTTaxMA0GCWCGSAFlAwQCAQUABCBA2JSI5aohNc4h
/65kOpGYoS9NYapVvSGzm9e011AYgQUdoD4rfCb4CrIsxlzQqZ6PgXMow8CAicQ"

```

},
"password": "Qwerty123",
"templateId": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
}

```

- Значение поля `request.data` в запросе можно получить, выполнив команду:

```
openssl base64 -in operator.p12 | tr -d '\n'
```

где `operator.p12` – файл сертификата.

1.26.2 Пример ответа на обращение к методу

```

{
  "status": 200,
  "data": {
    "serialnumber": "5d53f9f8c9d192b77fa4d0d23026d48fb69a48cb",
    "fingerprint": "9678302bd909deef55bcf4fbc4e30bfd8236a8a9",
    "name": "operator",
    "issuerDN": {
      "CN": [
        "RootCA"
      ],
      "OU": [
        "Department"
      ],
      "O": [
        "organization"
      ],
      "L": [
        "City"
      ],
      "DC": [
        "Component"
      ],
      "C": [
        "RU"
      ]
    },
    "subjectDN": {
      "CN": [
        "operator"
      ]
    },
    "subjectAltName": {},
    "validFrom": "2025-01-28T12:17:51Z",
    "validTo": "2027-01-28T12:17:51Z",
    "privateKey": {
      "contentType": "application/octet-stream",
      "fileName": "operator.key",
      "data":
        "LS0tLS1CRUdJTiBSU0EgUFJJJVkFURSBURVktLS0tLQpNSU1FcFFJQkFBS0NBUEVBenJMWlUxUFRSN1lPMXhs
        bjQzTmFDeXd4RnpKeGMvS2pLU1BXc01Qd05FVDZWTGVQCmlBUDkySWZhZnp0S1Y3VU9IMFIvdm5aQjN2cjF6b
        jJoYmtNMHRlZ0FNL1BtVGtCM3dDVUlMSTN2Y1FjRDR0S1gKb29DZ3JCUExrTDdFODFRTkdPbDBwTDZ6QkdVMm
        w5TGxqa0h2K1NFU2hlUE5VWEhJanptL0tQOEhlMWliRDUwSgpyZXl5dDErSFZlUXFEOWxtdeEtVdkh6NGlkOFg
        1OTRmVFclL3QyTy9KbUkwS2Z3aVRvdXlhSVJGeGNJcEt3NXQxCjZRSWtoSFBUUWtVMXcyWE1kTDJ3ZXVrUEVh
        UGU2N2hpMWglSHh4WW1OMnFabGVWWDNuVTM1Vm5sd2Z3WW9PM1AKdStocmUxL20yTDRVYW9tWW1OVHJ4L3VsT

```

```

WlnNy8vZUozaTU1MlFJREFRQUJBb0lCQUJoTmPkRTIrTt15c2ttKwpZZGtyejgxaXZxcGpNVlVBVlVnUUI3VU
1zc1pZbWlUaDJNL1ZwVkfEYURDL3ZrWForVTA2bXZrQVUwNG1ESWNBci9UMDNUSEFyZFRPbWxLQ0VncmYrT3J
WWFpHenlGZFpDWkZDclprZlPRM3p6T3ptL2V4TS81c20wak55aVVHVVIKUKZHaHhKYL1JXUW5TVH1xcGNkc0tQ
NFJRMlJhNnpneUJZVGM0UddCeXFyLzZWcGxFTEJEdC9kUVp6YmY2T3FZYQpWaEd0REtqTDlmdVhGmMnadFlye
jh2ODFqWklzYi9WYVRFNdhSaVR4NlBPULVvTHYzanRBZ2Rtd1ArN3RwbU5LC1NiS3o3QnBtOGR3VDBRTXBHUW
d3ZDhCS3BMAVa4YlBoZC8xcm1pTzgvUzgwS2lpdnN6eFdKZUhGVWdMWWNNMTcKbUV4V31zRUNnWUVBM1FZQnZ
TcWo0Q3pLK1NxeXlDT1p1UddQb3hCYjJ2SXkxdzhHYk1jc2Q2Z2M0U1RNdEwvQwppZUNSVW8vSec0WGRETxhF
cnBNV0VIRXN6dHZqWFQrc08zNDJlZGhTNExObW0rMEgvSGcrOXREVRHWQldjenFUCk1CRkUrSzhFRW5OM1pNV
1J3bWxiTzd3TTh2MkUwRzBNSlkyc2htRTERNWNkQVNpb1RkWUhZSmtDZ1lFQTcyAUyKZ0hIUghJaFZrVfHxZ2
1na201aTBWampoaTlKdnk4QWVzT1RlMDFzWF1QcEpZdUJKOHbWDRaQ3FjV2FacjYxVgo3ejBtOVBGVnBCZkN
0UnAzCjRUSmg4aG5LTEdSRytDYzIyMHb3d2RrZHZDcWRUUG5OdVFNdUx0Qi9LMnc3d1BzCk5XeJZHSnZxdytq
dTErNURJREDjendqNFUyUldHckE3Qm1ySGEwRUNnWUVBbk3aXNISUJ2b3M4QXG3aG9LN0cK0XfSWHzyT1p2U
ThPUUJPZWczTFolckFvTks4ZENYTkRnNUTVblp6R2RwSnRyK0VKVfHkV2k3TC9BRS8xVDFvSQp6K05wQnduYU
90dlk5SjREcUN3c3ZYdlNXbjVibml1Qmk1ZHYrWTlQSFJxNUdhSnd5dnNCNnVBYmRBdHNCbzhYCnFZb2ZDTmp
XK2kxaUdNEp4REswNnhFQ2dZRUF4TUZzL3B3S2hZeWfia1lXWG1pQ3RaSHhtKzhSaHaXREhmdmQKL1ZtTHU3
NWdYm1l6aTl4elNoek8wNlhXUG02cEhoVm5OFhCVDR0TDdCbE8veElxalFpZnJfVktHc3BNVE5IegpkKzZtW
k0rZGhKYlVNSkNnbEo4bHlXaTbtWnFTRUUXMEFwQjJsbVVGczA2cmhMRWZYOFVHYjZ5YwtjYWpjVG1tCnVPWm
0wRUVDZ1lFQWxHaEhuVVEwSmxsYWhocS8rMWRBWEaL1BjQ3BDUTR4M2w4NXU2bnpKcVRTbDNBa0RPWjEKOu0
2ZytYK2RIUEh0UWFDclNVVEpUYXdmmandKMDB2bXg0NTdrbZBGYkgxWFRNSENTcVQ5RWxIaklUZGhWZwt1cQp0
TE9WWU9GV2t1sZrLUhPDRnZvTTVyR0lRTTlSVGJQaTdsM2RrS1RKsXFMKzF0RVovQXhUQTJZPQotLS0tLUVOR
CBSU0EgUFJJVkfURSBRLVktLS0tLQo="
},
"certificate": {
  "contentType": "application/octet-stream",
  "fileName": "operator.pem",
  "data":
"LS0tLS1CRUdJTiBDRVJUSUZJQ0FURSB0tLS0tCk1JSUVsRENDQW55Z0F3SUJBZ01VWFZQNStNblJrcmQvcE5E
U01DYlVqN2FhU01zd0RRWUpLb1pJaHJzTkFRRUWkQlFBd2RERUXNQWtHQTFVRUJoTUNVbFV4R1RBWEJnb0pra
WFKay9JclpBRVpGZ2xeyJixd2IyNWxib1F4RFRBTApcZ05WQkFjTUJFTnBkSGt4R1RBVEJnTlZCQW9NREc5eV
oyRnVhWHBoZEdsdmJqRVRNqkVHQTFVRUN3d0tSR1Z3C1lYSjBiV1Z1ZERFUE1BMEdBmVVFQXkd3R1VtOXZkRU5
CTUI0WERUSTFNREV5T0RFeU1UYzFNVm9YRFRJm01ERXkKT0RFeU1UYzFNVm93RXpFuk1BOEdBMVVFQXkd3SWIz
QmxjbUyYyJnJd2dnRWlNQTBHQ1NxR1NJYjNEUUVVCQVfVQpBNElCRHdBd2dnRUtBb0lCQVFEt3N0bFRVOU5Ic
Gc3WEZHmpjMW9MTERFWE1uRno4cU1wRTlhd3cvQTBSUHBVCnQ0K0lBLzNZaDlWl08wcFh0UTRMUkgrK2RrSG
UrdlhPZmFGdVF6Uze2QUF6OCtaTlFIZkfkUWg4amU5eEJ3UGkKMgXlaWdLQ3NFOHVRdnNUelZBMfK2WfNrdnJ
NRVpUYVgwdVdPUWUvNUlSS0Y0ODFSY2NpUE9iOG8vd2Q3V0pzUApUW10N0xLM1g0ZfVqk29QMLdhMHBTOGZQ
aUozeGZuM2g5TmJuKzNZNzhtWWpRcC9DSk9pN0pvaEVYRndpa3JECm0zWHBBAvNFYzhsQ1JUWERaY2gwdmJCN
jZROFJvOTdydUdMV0hrZkhGaVkyXBTvJvWZmVkvGZsV2VYQi9CaWcKN2MrNzZHdDdYK2JZdmhScWlaaVktT3
ZIKzZVeUteDi85NG5lTG5uWkFnTUJBQUdqZnpCOU1Bd0dBMVVKRXdfQgovd1FDTUFBd0h3WURWUjBqQkjd0Z
vQV6ZlNoYURXY254cW1laTA2djg0YXFHNHvYzh3SFFZRFZSMGxCQl13CkZBWU1Ld1lCQlFVSEf3SudDQ3NH
QVFVRk1J3TUVNqjBHQTfVZERnUVdcQl1F3UetoQWd0Q1GbeJ0REhVZGpUbnYKWC9KMLdEQU9CZ05WSFE4QkFmO
EVCQU1DQmVBd0RRWUpLb1pJaHJzTkFRRUxUUFEEZ2dJQkFDWWovTWk0WkZxQQpBOUd0Q1EwN1Z6dDFPMGRjVW
t5bVhLOUZOVUhYNEQ0SUFOZlAva2Yrdn16YzVrd0Y4YVI5VzZKcTh6dU50OUlwCnpwN1lGQXl2UWpoeUJ4YwV
5Q2d4dkdVdFZsYmVZUE9FZ2RjenhoQ2hyMUc4YTN3S0JPcWd0bzZocnhvZGlkd0IKbHVzdHA2dHhLUGRoQVBO
RXRvU2lKR2laZz1RZm5OakVknGU3eXhGaXFHQTRJQkZVSUZhYVpEeW90VDVPnlRFRQpWaUY0cTloUVY0OHk2a
kJYYWlrvKZtdGpjRXUYSmRESmpmb3JOVkkMcDNkdGFzT2Q0QUJRv0pPbFZFSGQzL0FGClRCR0ZFWC94bHRZN3
16NDQwU0prRm45TXo4bytjbkM0RH15TkkxQWki4N2RWRWZ3U0hhWVM3RUtNbEVEVndUcU4KcXlQMURLOFJMv1V
taXBGUmolRmdwT0l0WnVwbGU0a3A1T2MwU2krL0RZWmpZZ0pxUnBsWVUWt1IxUHNONjRHbQo5Q05xN3ZnOE4x
S1FBVkvVRWHlsS1Bpcz1ORnYxWWtrMFErUnpTRGF4blRiRm9YU2tkd2g2Tk5tdUtwTHZYZ3VhCmVuUW1Ba2tpL
0g1bFNadDRqVWhvOHQwZfJyV2k5U09yRGxEVU9WZki3ZWfJek1lNed4WfdBeG1xUmZnelhYRk0KcUZ5SskprQU
puUW1MdHNmb3EvT0RzaWJibnlpZ1RHUWNjT0J6bWRPU1MwZEFKZCtLTkhNVUxtUuJTYTlXt0xjRgplVnBBdzM
1N3VKLysxU3ZaZnlGckZ0WXBscFN2L0ZzR2tUZFpHWm53Z1FJYm1QU0p6Vct2Q1BTv3B1V5YVowCmdNczNO
WHZwS2dzblFPT0JlYW1zZ1MxaW9DV1U0ZDUxci0tLS0tRU5EIEFUF1RJRklDQVRFLS0tLS0K"
},
"keyUsages": [
  {

```

```

    "id": "31a1b2a8-e197-47e4-8cd7-2d463f83166f",
    "code": "DIGITAL_SIGNATURE",
    "description": "Цифровая подпись",
    "value": "Digital signature"
  },
  {
    "id": "519e5b6c-9d21-4478-97dd-f0e09c593151",
    "code": "NON_REPUDIATION",
    "description": "Подтверждение подлинности",
    "value": "Non-repudiation"
  },
  {
    "id": "4973c634-29a2-4f6a-903e-e6b2313ab0eb",
    "code": "KEY_ENCIIPHERMENT",
    "description": "Шифрование ключей",
    "value": "Key encipherment"
  }
],
"extendedKeyUsages": [
  {
    "id": "b170dc22-c502-4c90-800c-d883f2105c44",
    "oid": "1.3.6.1.5.5.7.3.2",
    "value": "Аутентификация клиента",
    "description": "Сертификат может использоваться при установлении защищенного соединения по протоколу TLS для подтверждения подлинности клиента.",
    "created": "2024-07-09T18:56:51.928948Z",
    "updated": "2024-07-09T18:56:52.118579Z",
    "default": true
  },
  {
    "id": "0848b618-2a61-42f8-be8d-192b87d73f92",
    "oid": "1.3.6.1.5.5.7.3.4",
    "value": "Защита электронной почты",
    "description": "Сертификат может использоваться для защиты электронной почты (подпись, шифрование, соглашение о ключах).",
    "created": "2024-07-09T18:56:51.928948Z",
    "updated": "2024-07-09T18:56:52.118579Z",
    "default": true
  }
],
"ca": false
}

```

1.27 Метод расшифровки сертификата

POST API – Расшифровка сертификата	
Метод доступен администратору	
URL – certificate-authority-service/api/v2a/public/parse/pem	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A контейнеры сертификатов/parse_2	
Query	
Request	
{	

request: {	Файл сертификата
contentType (string) [опционально],	Тип загружаемого файла (HTTP MediaType) - application/octet-stream)
fileName (string) [опционально],	Имя загружаемого файла
data (string:binary)	Содержимое загружаемого файла (массив байт в Base64)
}	
}	
Response	
{	Ответ JSON в HTTP-body
id (UUID),	Идентификатор сертификата
chain: {	Цепочка сертификатов (рекурсивный объект)
id (UUID),	Идентификатор сертификата
name (string),	Имя сертификата (на основе CN)
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
issuer: {...}	Издатель сертификата (вложенный объект)
},	
serialnumber (string),	Серийный номер сертификата
fingerprint (string),	Фингерпринт сертификата
name (string),	Имя сертификата (на основе CN)
issuerId (UUID),	Идентификатор издателя сертификата
issuerDN: {	Имя субъекта издателя сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта издателя из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
subjectAltName: {	Альтернативное имя субъекта сертификата
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key - один из перечисленных в enum параметров, а value - значение параметра
},	
keyAlgorithm (enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN),	Алгоритм ключа сертификата ЦС

hashAlgorithm (enum: SHA1, SHA256, SHA384, SHA512, SHA3_256, SHA3_384, SHA3_512, RSASSA_PSS, MD5, GOST_R_34_11_2012, UNKNOWN),	Алгоритм подписи сертификата ЦС
keyBits (int32),	Длина ключа сертификата ЦС
templateId (UUID),	Идентификатор шаблона
templateName (string),	Имя шаблона
type (enum: CERTIFICATE, ROOT_CA, SUB_CA, UNKNOWN),	Тип сертификата
validFrom (instant),	Дата начала действия сертификата (ISO 8601)
validTo (instant),	Дата окончания действия сертификата (ISO 8601)
status (enum: ACTIVE, HOLD, REVOKE, REQUEST, UNKNOWN),	Статус сертификата
isValid (boolean),	Флаг: сертификат действует
isExpired (boolean),	Флаг: сертификат истек
actions: {	Доступные действия по выгрузке
p12 (boolean),	Флаг: выгрузка pkcs12
csr (boolean),	Флаг: выгрузка pkcs10
pem (boolean)	Флаг: выгрузка сертификата
},	
publicKey (string),	Открытый ключ
certificateType (string),	Тип сертификата (X.509)
version (int32),	Версия сертификата
subjectKeyIdentifier (string),	Идентификатор ключа сертификата
authorityKeyIdentifier (string)	Идентификатор ключа издателя сертификата
keyUsages: {	Назначение ключа сертификата
id (uuid),	Идентификатор элемента
code (enum: DIGITAL_SIGNATURE, NON_REPUDIATION, KEY_ENCRYPTMENT, DATA_ENCRYPTMENT, KEY_AGREEMENT, KEY_CERT_SIGN, CRL_SIGN, ENCRYPT_ONLY, DECRYPT_ONLY, UNKNOWN),	Перечисление использования ключа
value (string),	Наименование элемента
description (string)	Описание использования ключа
},	
extendedKeyUsages: {	Расширенное назначение ключа сертификата
id (uuid),	Идентификатор назначения
value (string),	Наименование элемента
oid (string),	OID назначения
description (string),	Описание использования ключа
updated (instant),	Время обновления (ISO 8601)
created (instant),	Время создания (ISO 8601)
default (boolean)	Флаг: расширенное использование по умолчанию
},	Описание OID
ca (boolean),	Флаг: сертификат ЦС
revocation: {	Сведения об отзыве сертификата ЦС
date (instant),	Дата отзыва
number (int4),	Код причины отзыва
value (string)	Значение причины отзыва
},	
aiaUrls (string[]),	URL AIA
ocspUrls (string[]),	URL OCSP
crlUrls (string[]),	URL CRL
deltaCrlUrls (string[]),	URL Delta CRL
userId (uuid),	Идентификатор субъекта
subjectId (uuid),	Идентификатор субъекта
created (instant)	Время создания (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
}	


```

    "id": "473eee90-bb99-4831-9772-1a26fb0f9ed2",
    "name": "SUB_CA_INFORM",
    "subjectDN": { "CN": ["SUB_CA_INFORM"], "O": ["test"] },
    "issuer": {
      "id": "4489978c-2941-4a37-8948-8b422c66c8d6",
      "name": "aladdin",
      "subjectDN": {
        "CN": ["aladdin"],
        "OU": ["Java555"],
        "O": ["aladdin"],
        "L": ["Novosibirsk"],
        "DC": ["Java555"],
        "C": ["RU"]
      },
      "issuer": null
    }
  },
  "serialnumber": "7b0f69dd4500f8fd1035688170c1458505636a24",
  "fingerprint": "865715305229bebcd43ea0887bbcc6137be11145",
  "name": "Dena Marshall",
  "issuerId": "473eee90-bb99-4831-9772-1a26fb0f9ed2",
  "issuerDN": { "CN": ["SUB_CA_INFORM"], "O": ["test"] },
  "subjectDN": { "CN": ["Dena Marshall"] },
  "subjectAltName": {
    "RFC822NAME": ["Thenterage87@test.local"],
    "MS_UPN": ["Thenterage87@test.local"]
  },
  "keyAlgorithm": "RSA",
  "hashAlgorithm": "SHA256",
  "keyBits": 1024,
  "templateId": "aa03e458-50cd-46b8-82cd-d5612ed3b647",
  "templateName": "Smartcard Logon",
  "type": "CERTIFICATE",
  "validFrom": "2024-10-28T12:28:38Z",
  "validTo": "2026-10-28T12:28:38Z",
  "status": "ACTIVE",
  "isValid": true,
  "isExpired": false,
  "actions": { "p12": true, "pem": true, "csr": false },
  "publicKey": "Sun RSA public key, 1024 bits\n  params: null\n  modulus:
1196101928316451389921208433420568117961940018904587011928779977021239941914786245485
6256868949150718470293757440760506112111583590465580628771037253406310228702962084237
4068436641309113741361277296919163128760687231220570652198675703806970445639104821549
716138753463094951540551128680075950344089038867665689\n  public exponent: 65537",
  "certificateType": "X.509",
  "version": 3,
  "subjectKeyIdentifier": "cfe8180f778233ead97de2e61bd0b17bb2637b88",
  "authorityKeyIdentifier": "5dbbddd274575b47750b47f2044d58519cb3558f",
  "keyUsages": [
    {
      "id": "31a1b2a8-e197-47e4-8cd7-2d463f83166f",
      "code": "DIGITAL_SIGNATURE",
      "description": "Цифровая подпись",
      "value": "Digital signature"
    }
  ]
}

```

```
    },
    {
      "id": "519e5b6c-9d21-4478-97dd-f0e09c593151",
      "code": "NON_REPUDIATION",
      "description": "Подтверждение подлинности",
      "value": "Non-repudiation"
    },
    {
      "id": "4973c634-29a2-4f6a-903e-e6b2313ab0eb",
      "code": "KEY_ENCIPHERMENT",
      "description": "Шифрование ключей",
      "value": "Key encipherment"
    },
    {
      "id": "834d6f41-5d03-43e2-9b1c-82da036eed75",
      "code": "DATA_ENCIPHERMENT",
      "description": "Шифрование данных",
      "value": "Data encipherment"
    }
  ],
  "extendedKeyUsages": [
    {
      "id": "b170dc22-c502-4c90-800c-d883f2105c44",
      "oid": "1.3.6.1.5.5.7.3.2",
      "value": "Аутентификация клиента",
      "description": "Сертификат может использоваться при установлении защищенного соединения по протоколу TLS для подтверждения подлинности клиента.",
      "created": "2024-07-09T18:56:51.928948Z",
      "updated": "2024-07-09T18:56:52.118579Z",
      "default": true
    },
    {
      "id": "0848b618-2a61-42f8-be8d-192b87d73f92",
      "oid": "1.3.6.1.5.5.7.3.4",
      "value": "Защита электронной почты",
      "description": "Сертификат может использоваться для защиты электронной почты (подпись, шифрование, соглашение о ключах).",
      "created": "2024-07-09T18:56:51.928948Z",
      "updated": "2024-07-09T18:56:52.118579Z",
      "default": true
    },
    {
      "id": "9f40092b-d5e1-4c34-8912-45ddc6971f2e",
      "oid": "1.3.6.1.4.1.311.20.2.2",
      "value": "Вход с MS смарт-картой",
      "description": "Сертификат может использоваться физическим лицом для входа в систему с помощью смарт-карты.",
      "created": "2024-07-09T18:56:51.928948Z",
      "updated": "2024-07-09T18:56:52.118579Z",
      "default": true
    }
  ],
  "ca": false,
  "revocation": null,
  "aiaUrls": ["https://test-aia.ru"],
```

```

    "ocspUrls": ["DK20"],
    "crlUrls": ["https://test.ru"],
    "deltaCrlUrls": [],
    "userId": null,
    "subjectId": "000000a6-d0c3-4a4e-ae66-be5ad83a3ba2",
    "created": "2024-10-28T12:28:39.024349Z",
    "updated": "2024-10-28T12:28:39.024359Z"
  }
}

```

1.28 Метод расшифровки запроса на сертификат

POST API – Расшифровка запроса на сертификат	
Метод доступен администратору	
URL – certificate-authority-service/api/v2a/public/parse/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A контейнеры сертификатов/parse_1	
Query	
Request	
{	
request: {	Файл сертификата
contentType (string) [опционально],	Тип загружаемого файла (HTTP MediaType) – application/octet-stream
fileName (string) [опционально],	Имя загружаемого файла
data (string:binary)	Содержимое загружаемого файла (массив байт в Base64)
}	
subjectName: {	Поля отличительного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectName, переопределяют значения соответствующих полей SDN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET, NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	
}, [опционально]	
subjectAltName: {	Поля альтернативного имени субъекта, которое должно попасть в сертификат. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра. Значения полей, указанные в subjectAltName, переопределяют значения соответствующих полей SAN запроса на сертификат, при условии, что они соответствуют значениям атрибутов субъекта.
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	
} [опционально]	
}	
Response	
{	Ответ JSON в HTTP-body
name (string),	Имя сертификата (на основе CN)
subjectDN: {	Имя субъекта сертификата
(enum: CN, UID, E, EMAILADDRESS, MAIL, SN, GIVENNAME, INITIALS, SURNAME, OU, O, L, ST, DC, C, UNSTRUCTUREDADDRESS, UNSTRUCTUREDNAME, POSTALCODE, BUSINESSCATEGORY, TELEPHONENUMBER, PSEUDONYM, POSTALADDRESS, STREET,	Поля разделенного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра

NAME, T, DN, DESCRIPTION, INN, OGRN, OGRNIP, SNILS, INNLE, UNKNOWN): string[]	
},	
subjectAltName: {	Альтернативное имя субъекта сертификата
(enum: RFC822NAME, DNS_NAME, IPADDRESS, DIRECTORY_NAME, UNIFORM_RESOURCE_ID, REGISTERED_ID, MS_UPN, MS_GUID, KRB5PRINCIPAL, PERMANENT_IDENTIFIER, XMPP_ADDR, SRV_NAME, SUBJECT_IDENTIFICATION_METHOD, UNKNOWN): string[]	Поля альтернативного имени субъекта из сертификата. В формате key-value. Где key – один из перечисленных в enum параметров, а value – значение параметра
},	
keyAlgorithm (enum: RSA, ECDSA, GOST_R_34_10_2012, UNKNOWN),	Алгоритм ключа сертификата ЦС
keyBits (int32),	Длина ключа сертификата ЦС
subjectKeyIdentifier (string)	Идентификатор ключа сертификата
}	

1.28.1 Пример использования метода

```
curl -k -X 'POST' \
  'https://192.168.111.100/certificate-authority-service/api/v2a/public/parse/pkcs10' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -H 'Cookie:
token=eyJhbGciOiJIc2E1MTJ3aXR0eUlNbIn0.eyJzZXNzaW9uSWQ0iOiI1ZDaxMDQwZS04NjM2LTRjNTgtYWJ1YilhYTdjZTEyYzRiNWUuIiLCJpYXQiOiE3MzgwNjQ1NzAsImV4cCI6MTczODA2NDc1MH0.tgOSvc3MWwQw07grohKe_ixDqVHS_bedcBCNJkFxGg4vAxf39QP1wHjEuKPEC1BT1bvP8jGfTT1cSx1bvhPnF0q_HUumv-C_wRJj4UgKmv1XrcoY6ceyD62ITkdQ-x1zmzr0hp-XAMRQ5Y6H3NsJNloT-koEUQiCI18CYvmAZUz_mvmpmtULchOsfGXykFay1YPIsR2JEEAaf4fRRU5RZIWqIBj6bjcuNAiVhtdJ0hmgS0RPZcR514vpqIQQ-RaOyOD0axeIepN0Yixz9QeL2ZmpHj7gXxnFYWG9uslsizYFzQt34afYgmq7Dp-OxAS5Yc5u095V9kTTbOyfSMLXaLRg' \
  -d '{
    "request": {
      "data":
"LS0tLS1CRUdJTiBDRVJUSUZJQ0FURSBzRVFVRVNULS0tLS0KTU1JQ3RqQ0NBYUFDQVFBd0V6RVJNQThQTFFV
RUF3d01WR1Z6ZERnM01UY3dnZ0VpTUEwR0NTcUdTTSWIzRFFFQgpBUVVBQTRJQkR3QXdnZ0VLQW9JQkFRQzc5R
EtJaVQvc3gyTEVucFEExWHVNcHU4MktpeTVvUlFvQTZyMHdjN1BjCmcYWFJJM01xeHZYSjdLSWFXVWpob2FSNF
1Dak4vcD15Z1RMQm0rSnJBSDNNdktWV2w0Z1plb3hNYW1Yb2xvFVnEKT1JYW55qckczVCtZMkdaWStIM0hyMFB
tSWFwODY4NFR1U3JMNzd6OGVlNHpnTETEd05jd29Ud3g3dVR3K1RScApUSzYzdkZTeUcxQ253QmNacXR0Y2dw
bmwzTkhkZW5Rci9tZ2ZzU0dSszJLUU85U01zTFBQNN0ZitOaG5iaHdwCm04dEtBNm9RQ1pjaTFKQytYdExLZ
ENLWFh1QTdjeXZDd3ZBRWhBeW16b05OMFNIUktjbC9XeUMxNDhDVlh2SkckTHAvTnR2Z0ZlMVFOcVJUMEZSUU
xvaEVuZ1ROKzhneWVUeGZiWGxLeDhnM25BZ01CQUFHZ11EQmVCZ2txaGtpRwo5dzBCQ1E0eFVUQ1BNQTRHQTFF
VZER3RUIvd1FFQXdJQUFEQTlCZ05WSFJFRU5qQTBnUkIwWlhOME9EY3hOMEJ0C1lXbHNMbkoXb0NBR0Npc0dB
UVFCZ2pjVUFnT2dFZ3dRZEdWemREZzNNVGRBYldGcGJDNXlkVEFMQmdrcWhraUcKQXcwQkFRVURnZ0VCQUUzR
Vd0QUJWV00yN1ZKNmtDLzNRYXJLemxMZ2h4c090Ky8yMStsNmpCOVBwOFZtR1BYUgo1VFY0UkwrTDkyWWNuQk
tDU1Z3Y2N1aDJeEEeXRdQ0V0hYZ2o5UXRPZ0h3dGgzTkZkQVF6dUEwMTRiNk1BVz1FCmVick5LTWFhY1ZUR1J
BbmG2WldkVFZQSjRacTNaRURIMTNVK0JLU2E0M3FOeVp3QjBIAcTkc3czVz1WbmFsMEMKQ0Z3djM4NEVyam9p
TU92ZytGT0t1MThTty9lWm16cWN0M0IxZzIrSthuaGJMYklETk5iR1FMby9KOGxZS084Uwp1WnE0N3pxS2kvQ
kdPTmJaeFphSjRly0xN1JVvZyAyTWsrMmc3NTRTVG8xbC9ycTc0WW1iU2E0S3FLY25UZDJXCmtNMExpNkN3YU
p1YkQ0UE9TT3N5c3RERmdsTXBMbGpPd2lZPQotLS0tLUVORCBDRVJUSUZJQ0FURSBzRVFVRVNULS0tLS0K"
    }
  }'
```

- Значение поля `request.data` в запросе можно получить, выполнив команду:

```
openssl base64 -in req.csr | tr -d '\n'
```

где `req.csr` – файл запроса на сертификат.

1.28.2 Пример ответа на обращение к методу

```
{
  "status": 200,
  "data": {
    "name": "Test8717",
    "subjectDN": {
      "CN": [
        "Test8717"
      ]
    },
    "subjectAltName": {
      "RFC822NAME": [
        "test8717@mail.ru"
      ],
      "MS_UPN": [
        "test8717@mail.ru"
      ]
    },
    "keyBits": 2048,
    "keyAlgorithm": "RSA",
    "subjectKeyIdentifier": "a642079de5f054ab12f8ce2933dc9378fd57524d"
  }
}
```

МЕТОДЫ ЭКСПОРТА ФАЙЛОВ

1.29 Метод получения сертификата по идентификатору сертификата

GET API – Получение сертификата по идентификатору сертификата	
Метод доступен администратору и оператору при наличии полномочий	
URL – export-service/api/v2a/public/export/certificates/{certificateId}/certificate	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер экспорта/findCertificateById	
Query	
{	
certificateId (UUID)	ID сертификата
}	
Request	
-	
Response	
ResponseEntity->byte[]	

1.29.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/export-
service/api/v2a/public/export/certificates/24a9a005-129a-4045-a8dd-
296699ff596d/certificate' \
--
header 'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQ6I0I0MTcyODI2My0yY2VhLTQ4MT
EtOTZjMS1kMmZlNmI4YzA5MjEiLCJpYXQiOiE3MDIyNzk1MDAsImV4cCI6MTcwMjI3OTY4MH0.eMonQVtPflL6
91nD89VmB1Vvz6CK1Trnp0GdZJ5HQ2_EfrLDv-
THZzPfl1TG6SA_gQuqh1_N8f6njflqBqgM6PPGFN8FfLzHCV5feESazNaFDszOu90OFHy4p5fiRG9Ms-
_2U5xrl-DNHJ7hkOJNooAoa71ntVpG9Kk8kF-
EUVi7NKLsBFhQECHPzyhTopNAltNLo85TeVCbFlH064DLfFTTaFt5RRtvfBygkBzyeYpdHW_ISrVv-
IzmE2xd3xuJgaL06gtFnzcxErvBBDUKL603aZ6IWXGVIFspOxAO3U1J8bc5x15IhOJmasBPGwdrVcL9lWmOrU
TGzcd3EDYUA '
```

1.29.2 Пример ответа на обращение к методу

```
Issuer: CN=INITIAL_CA
Subject: CN=INITIAL_ADMIN
-----BEGIN CERTIFICATE-----
MIIDgDCCAmigAwIBAgIUIdwuUS1CDUiim29qeFV4ARk6ueowDQYJKoZIhvcNAQEL
BQAwFTEtMBEgAlUEAwKKSU5JVE1BTf9DQTAeFw0yMzEyMDgwNzI3NDZaFw0yNTEy
MDcwNzI3NDZaMBGxGjAUBgNVBAMMDU1OSVRJQUxfQURNSU4wgGEiMA0GCSqGSIb3
DQEBAQUAA4IBDwAwggEKAoIBAQQDZCbV6JagT064km6C1Aa7j9i2MiH4jptsCWGby
GIsvvs1LE/GBcTT4jziej7vMEY4S15cUW5UgQMwI2mxc9GnImmSj3veGwU7NFR6F
HWK+3T8Roa9c+omW+LEf+IAaZPA3GEerhEMZcgvetWbxXg79/a0aVJJcAQINo3nD
WmXpOnZEnjwtpK5BPLdNk3OnjhucA5slPNJRLrUby+hhww7GYD5f4TRGMEesq/DW
9cFYpga7oMDs7tZ1MJH8QAa7z3wVtx7q8V5pox6fyQ8kQDtwN9dwBmSp1gNrOQBx
jhn8Uk7PEffQ56F84/9GRKnMFVq4V2t1D1klDnZ1UxjyT4HhAgMBAAGjgcQwgcEw
NwYDVR0RBDAwLoENSU5JVE1BTBEBBRE1JTqAdBgorBgEEAYI3FAIDoA8MDU1OSVRJ
QUxhQUURNSU4wDgYDVR0PAQH/BAQDAgXgMB0GA1UdJQQWMBQGCCsGAQUFBwMCBggr
BgEFBQcDBDAfBgNVHSMEGDAwBSwD59XAM1X3/1TrhIUslYnnHGYRTAdBgNVHQ4E
FgQU2jx/3M5JuXw5kf1e0bPBbqIKMPEwCQYDVR0fBAIwADAMBgNVHRMBAf8EAjAA
MA0GCSqGSIb3DQEBCwUAA4IBAQBCV4ch1PWIxc5tNRe6qSLqsLmG7FNqeUIXk2qA
t+zd9mkXdfYS1jd0aSOflm68AQ2+Fx/Wv9udFoM+q1TBx6Wh6EliuUWxRctXe9qf
gYnlo3BcLSuApqWMqK7Wqv9WKU+243fiKFreXlCACZ410vG8xrjCN3JYtXb6W9eY
```

```
4ej2ZdYv6H3FoGclvj871l/X30EP4ElUYloJT1Ody6FTxidKF12yOV7NuC7xH7+Jy
VnUQwfVkJGhnVvaVq/gqkgrq7/nqy5COj1Rk0dc0n38WAAMP1M1L95C73InfE795W
HPvgT7G7MFnPTvbsGld4K694crUiFmMagLQOqmv91GD2ihw3
-----END CERTIFICATE-----
```

1.30 Метод получения запроса на сертификат по идентификатору сертификата

GET API – Получение запроса на сертификат по идентификатору сертификата	
Метод доступен администратору и оператору при наличии полномочий	
URL – export-service/api/v2a/public/export/certificates/{certificateId}/pkcs10	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер экспорта/findRequestById	
Query	
{	
certificateId (UUID)	ID сертификата
}	
Request	
-	
Response	
ResponseEntity->byte[]	

1.31 Метод получения цепочки сертификата по идентификатору сертификата

GET API – Получение цепочки сертификатов по идентификатору сертификата	
Метод доступен администратору и оператору при наличии полномочий	
URL – export-service/api/v2a/public/export/certificates/{certificateId}/chain	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер экспорта/findChainById	
Query	
{	
certificateId (UUID)	ID сертификата
}	
Request	
-	
Response	
ResponseEntity->byte[]	

1.32 Метод получения контейнера PKCS#12 по идентификатору сертификата

GET API – Получение контейнера pkcs12 по идентификатору сертификата	
Метод доступен администратору и оператору при наличии полномочий	
URL – export-service/api/v2a/public/export/certificates/{certificateId}/pkcs12	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер экспорта/findP12ById	
Query	
{	
certificateId (UUID)	ID сертификата
}	
Request	
-	
Response	
ResponseEntity->byte[]	

1.33 Метод получения сертификата Центра сертификации по идентификатору Центра сертификации

GET API – Получение сертификата по идентификатору Центра сертификации	
Метод доступен администратору	
URL – export-service/api/v2a/public/export/certificate-authorities/{caId}/certificate	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер экспорта/findCertificateByCald	
Query	
{	
caId (UUID)	ID ЦС
}	
Request	
-	
Response	
ResponseEntity->byte[]	

1.34 Метод получения цепочки сертификатов Центра сертификации по идентификатору Центра сертификации

GET API – Получение цепочки сертификатов по идентификатору Центра сертификации	
Метод доступен администратору	
URL – export-service/api/v2a/public/export/certificate-authorities/{caId}/chain	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер экспорта/findChainByCald	
Query	
{	
caId (UUID)	ID ЦС
}	
Request	
-	
Response	
ResponseEntity->byte[]	

1.35 Метод получения CRL по идентификатору Центра сертификации

GET API – Получение CRL по идентификатору Центра сертификации	
Метод доступен администратору	
URL – export-service/api/v2a/public/export/certificate-authorities/{caId}/crl	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер экспорта/findCrlByCald	
Query	
-	
Request	
-	
Response	
ResponseEntity->byte[]	

1.36 Метод получения DeltaCRL по идентификатору Центра сертификации

GET API – Получение DeltaCRL по идентификатору Центра сертификации	
Метод доступен администратору	

URL – export-service/api/v2a/public/export/certificate-authorities/{caId}/delta-crl	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер экспорта/findDeltaCrlByCald	
Query	
{	
caId (UUID)	ID ИС
}	
Request	
-	
Response	
ResponseEntity->byte[]	

МЕТОДЫ РАБОТЫ С ТОЧКАМИ РАСПРОСТРАНЕНИЯ

1.37 Метод генерации и публикации CRL по идентификатору Центра сертификации

POST API – Генерация и публикация CRL по идентификатору Центра сертификации	
Метод доступен администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2aКонтроллер%3А Конфигурации CRL для ЦС/generate	
URL – publisher-service/api/v2a/public/certificate-authorities/{cald}/crl-configuration/generate	
Query	
{	
id (UUID)	ID ЦС
}	
Request	
–	
Response	
–	

1.38 Метод генерации и публикации CRL по идентификатору Центра сертификации (устаревший)

POST API – Генерация и публикация CRL по идентификатору Центра сертификации	
Данный метод является устаревшим и будет исключен из публичного API в последующих версиях.	
Метод доступен администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2aКонтроллер%3А Конфигурации CRL для ЦС/generateDeprecated	
URL – publisher-service/api/v2a/ui/certificate-authorities/{cald}/crl-configuration/generate	
Query	
{	
id (UUID)	ID ЦС
}	
Request	
–	
Response	
–	

МЕТОДЫ РАБОТЫ С ТОЧКАМИ ПОДКЛЮЧЕНИЯ И РЕСУРСНЫМИ СИСТЕМАМИ

1.39 Метод поиска зарегистрированных ресурсных систем

GET API – Поиск ресурсных систем	
Метод доступен администратору и оператору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A ресурсные системы/findAll_4	
URL – subjects-service/api/v2a/public/resources	
Query	
{	
id (uuid) [опционально],	Фильтр: ID ресурсной системы
securityGroupId (uuid) [опционально],	Фильтр: ID группы безопасности
subjectId (uuid) [опционально],	Фильтр: ID субъекта
organizationalUnitId(uuid) [опционально],	Фильтр: ID подразделения
search (string),	Фильтр: полнотекстовый поиск по отображаемому имени
isConnected(boolean),	Фильтр: подключенная ресурсная система
isDefault (boolean)	Фильтр: ресурсная система по умолчанию
}	
Request	
–	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string),	DN ресурсной системы
subjectsCount (int64),	Количество субъектов ресурсной системы
isConnected (boolean),	Флаг: ресурсная система подключена
isDefault (boolean),	Флаг: локальная ресурсная система
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.40 Метод получения ресурсной системы по идентификатору

GET API – Получение ресурсной системы по идентификатору	
Метод доступен администратору и оператору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A ресурсные системы/findById_4	
URL – subjects-service/api/v2a/public/resources/{id}	
Query	
{	
id (uuid) [опционально]	ID ресурсной системы
}	
Request	
–	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string),	DN ресурсной системы
subjectsCount (int64),	Количество субъектов ресурсной системы
isConnected (boolean),	Флаг: ресурсная система подключена
isDefault (boolean),	Флаг: локальная ресурсная система
updated (instant),	Время обновления (ISO 8601)

created (instant)	Время создания (ISO 8601)
}	

1.41 Метод полной синхронизации ресурсной системы

PUT API – Полная синхронизация РС	
Метод доступен администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a] Контроллер%3A ресурсные системы/synchronize	
URL – ldap-service/api/v2a/public/resources/{resourceId}/synchronize	
Query	
{	
resourceId (UUID)	ID ресурсной системы
}	
Request	
-	
Response	
-	

1.42 Метод поиска точек подключения

GET API – Поиск точек подключения к РС	
Метод доступен администратору и оператору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a] Контроллер%3A Точки подключения ресурсных систем/findAll_12	
URL – ldap-service/api/v2a/public/connection-points	
Query	
{	
id (uuid) [опционально],	Фильтр: ID точки подключения
resourceId (uuid) [опционально],	Фильтр: ID ресурсной системы
search (string),	Фильтр: полнотекстовый поиск по отображаемому имени
sortDirection (string) [опционально],	Направления сортировки (ASC;DESC)
sortBy (string[]) [опционально],	Список полей, к которым применяется сортировка
pageOffset (integer) [опционально],	Смещение от начала списка (пагинация)
pageLimit (integer) [опционально]	Ограничение на размер выборки (пагинация)
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID точки подключения
title (string),	Имя точки подключения
domainType(enum: SAMBA_DC, MS_AD, RED_ADM, FREE_IPA, ALD_PRO, ALT_DOMAIN, UNKNOWN),	Тип точки подключения
connectionAddress (string),	Адрес (хост) подключения
useTls (boolean),	Флаг: использовать TLS при подключении
baseDn(string),	BaseDN точки подключения
username (string),	Имя пользователя ресурсной системы
status (string)	Статус точки подключения
resourceId (UUID),	ID ресурсной системы
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.43 Метод получения точки подключения по идентификатору

GET API – Получение точки подключения по идентификатору
--

Метод доступен администратору и оператору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3А Точки подключения ресурсных систем/findById_11	
URL – ldap-service/api/v2a/public/connection-points/{id}	
Query	
{	
id (uuid)	ID точки подключения
}	
Request	
–	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID точки подключения
title (string),	Имя точки подключения
domainType(enum: SAMBA_DC, MS_AD, RED_ADM, FREE_IPA, ALD_PRO, ALT_DOMAIN, UNKNOWN),	Тип точки подключения
connectionAddress (string),	Адрес (хост) подключения
useTls (boolean),	Флаг: использовать TLS при подключении
baseDn (string),	BaseDN точки подключения
username (string),	Имя пользователя ресурсной системы
status (string)	Статус точки подключения
resourceId (UUID),	ID ресурсной системы
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.44 Метод частичной синхронизации точки подключения

PUT API – Частичная синхронизация точки подключения	
Метод доступен администратору и оператору при наличии полномочий	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3А Точки подключения ресурсных систем/synchronize_1	
URL – ldap-service/api/v2a/public/connection-points/{pointId}/synchronize	
Query	
{	
pointId (UUID)	ID точки подключения
}	
Request	
–	
Response	
–	

МЕТОДЫ ПОЛУЧЕНИЯ ВЕРСИИ СЕРВИСОВ

1.45 Метод получения версии сервиса обратной совместимости

GET API – Получение версии сервиса обратной совместимости	
Метод доступен неаутентифицированному пользователю	
URL – backward-compatibility-service/api/version	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/version-controller/getApiVersion_1	
Query	
-	
Request	
-	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
version (string)	Версия сервиса обратной совместимости
}	

1.45.1 Пример использования метода

```
curl -k --location 'https://172.17.152.213/backward-compatibility-  
service/api/version' \  
--header  
'Cookie: token=eyJhbGciOiJSUzUxMiJ9.eyJzZXNzaW9uSWQiOiJjMDJjNzQ5Ny0xMWJlLTQwZjItOTE1M  
y1jMzQzZDZmZGU3OTMiLCJpYXQiOiJlMzMDIyODA5NjgsImV4cCI6MTcwMjI4MTE0OH0.Ht3EAVBMgEKNvFqbTe  
ZrZR0J7Vi-  
FlAGgYeFrfhSxATQ6zGzP08lX8zqEAiRbZP2Hrn9_YYd3TISTmULuNBL1e1evBs74uALeSuOfC_tdlPDp1Z4X  
dxlWPSszW7XyqisYREBO7BxT0wW5s9rFiqSmHK_wiIL25oR2IR2JA--bzc2fUu-  
DCLQXhbR5qKARK8RjppDyeed49qTbOWhsmEI4D6UF10_FO-  
lnL7B0GoYXhL9eDlffs6uFQaOQYq_IF9fWCxlnBfhTvTk79HAIQtz16AIDBR0T40P96JYPm93Kvf0bj562-  
yz5-T_QcEdmquxMjv5txn0ln9W3wb7KWfoIQ'
```

1.45.2 Пример ответа при успешном использовании данного метода представлен ниже:

```
{
  "status":200,
  "data":
    {
      "version":"develop"
    }
}
```

Метод должен возвращать статус «502» в случае ошибки получения версии сервиса обратной совместимости.

Данный метод может быть использован для диагностики работоспособности всех сервисов программы, так как в случае отказа в обслуживании любого из сервисов (не только сервиса обратной совместимости) ответ на обращение к данному методу будет иметь статус «502».

1.46 Метод получения версии сервиса внешних интеграций

GET API – Получение версии сервиса внешних интеграций
Метод доступен неаутентифицированному пользователю
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/version-controller/getApiVersion

URL – external-integration-service/api/version	
Query	
–	
Request	
–	
Response	
ResponseEntity -> ItemResponse -> {	Ответ JSON в HTTP-body
version (string)	Версия сервиса внешних интеграций
}	

МЕТОДЫ РАБОТЫ С SYSLOG-СЕРВЕРАМИ

1.47 Метод поиска Syslog-серверов

GET API – Поиск Syslog-серверов	
Метод доступен только администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a] Контроллер%3A Syslog сервера/findAll_10	
URL – logs-service/api/v2a/public/syslog	
Query	
–	
Request	
–	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID Syslog-сервера
host (string),	Имя хоста Syslog-сервера
port (int32),	Порт Syslog-сервера
protocol (enum: UDP, TCP, UNKNOWN),	Протокол Syslog-сервера
active (boolean)	Флаг: состояние настройки публикации событий
}	

1.48 Метод получения Syslog-сервера по идентификатору

GET API – Получение Syslog-сервера по идентификатору	
Метод доступен только администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a] Контроллер%3A Syslog сервера/findById_9	
URL – logs-service/api/v2a/public/syslog/{id}	
Query	
{	
id (uuid)	ID Syslog-сервера
}	
Request	
–	
Response	
{	Ответ JSON в HTTP-body
id (UUID),	ID Syslog-сервера
host (string),	Имя хоста Syslog-сервера
port (int32),	Порт Syslog-сервера
protocol (enum: UDP, TCP, UNKNOWN),	Протокол Syslog-сервера
active (boolean)	Флаг: состояние настройки публикации событий
}	

1.49 Метод создания Syslog-сервера

POST API – Создание Syslog-сервера	
Метод доступен только администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a] Контроллер%3A Syslog сервера/create	
URL – logs-service/api/v2a/public/syslog	
Query	
–	
Request	

{	
host (string),	Имя хоста Syslog-сервера
port (int32),	Порт Syslog-сервера
protocol (enum: UDP, TCP, UNKNOWN)	Протокол Syslog-сервера
}	
Response	
{	Ответ JSON в HTTP-body
id (UUID),	ID Syslog-сервера
host (string),	Имя хоста Syslog-сервера
port (int32),	Порт Syslog-сервера
protocol (enum: UDP, TCP, UNKNOWN),	Протокол Syslog-сервера
active (boolean)	Флаг: состояние настройки публикации событий
}	

1.50 Метод обновления Syslog-сервера

PUT API – Обновление Syslog-сервера	
Метод доступен только администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3ASyslog сервера/updateById	
URL – logs-service/api/v2a/public/syslog/{id}	
Query	
{	
id (uuid)	ID Syslog-сервера
}	
Request	
{	
host (string),	Имя хоста Syslog-сервера
port (int32),	Порт Syslog-сервера
protocol (enum: UDP, TCP, UNKNOWN)	Протокол Syslog-сервера
}	
Response	
{	Ответ JSON в HTTP-body
id (UUID),	ID Syslog-сервера
host (string),	Имя хоста Syslog-сервера
port (int32),	Порт Syslog-сервера
protocol (enum: UDP, TCP, UNKNOWN),	Протокол Syslog-сервера
active (boolean)	Флаг: состояние настройки публикации событий
}	

1.51 Метод деактивации Syslog-сервера

PATCH API – Деактивация Syslog-сервера	
Метод доступен только администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3ASyslog сервера/deactivate	
URL – logs-service/api/v2a/public/syslog/{id}/deactivate	
Query	
{	
id (uuid)	ID Syslog-сервера
}	
Request	
–	
Response	
–	

1.52 Метод активации Syslog-сервера

PATCH API – Активация Syslog-сервера	
Метод доступен только администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a] Контроллер%3A Syslog сервера/activate	
URL – logs-service/api/v2a/public/syslog/{id}/activate	
Query	
{	
id (uuid)	ID Syslog-сервера
}	
Request	
–	
Response	
–	

1.53 Метод удаления Syslog-сервера

DELETE API – Удаление Syslog-сервера	
Метод доступен только администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a] Контроллер%3A Syslog сервера/deleteById_2	
URL – logs-service/api/v2a/public/syslog/{id}	
Query	
{	
id (uuid)	ID Syslog-сервера
}	
Request	
–	
Response	
–	

МЕТОДЫ РАБОТЫ С УЧЕТНЫМИ ЗАПИСЯМИ

1.54 Метод поиска учетных записей

GET API – Поиск учетных записей	
Метод доступен администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A Учетные записи/findAll_9	
URL – security-service/api/v2a/public/accounts	
Query	
{	
id (uuid[]) [опционально],	Фильтр: ID учетной записи
notId (uuid[]) [опционально],	Фильтр: ID учетной записи (исключающий)
logins (uuid[]) [опционально],	Фильтр: логин учетной записи
search (string) [опционально]	Фильтр: полнотекстовый поиск по отображаемому имени
}	
Request	
–	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID учетной записи
login (string),	Логин учетной записи
principalName (string),	Отображаемое учетной записи
role (enum: ADMINISTRATOR, OPERATOR, UNKNOWN),	Роль учетной записи
status (enum: ACTIVE, BLOCKED, UNKNOWN),	Статус учетной записи
created (instant),	Время создания (ISO 8601)
isLinked (boolean)	Флаг: связь с субъектом из ресурсной системы
}	

1.55 Метод получения учетной записи по идентификатору

GET API – Получение учетной записи по идентификатору	
Метод доступен администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a Контроллер%3A Учетные записи/findById_8	
URL – security-service/api/v2a/public/accounts /{id}	
Query	
{	
id (uuid)	ID учетной записи
}	
Request	
–	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID учетной записи
login (string),	Логин учетной записи
principalName (string),	Отображаемое учетной записи
role (enum: ADMINISTRATOR, OPERATOR, UNKNOWN),	Роль учетной записи
status (enum: ACTIVE, BLOCKED, UNKNOWN),	Статус учетной записи
created (instant),	Время создания (ISO 8601)
isLinked (boolean)	Флаг: связь с субъектом из ресурсной системы
}	

1.56 Метод получения учетной записи по отпечатку сертификата

GET API – Получение учетной записи по отпечатку сертификата	
Метод доступен администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a] Контроллер%3A Учетные записи/getByFingerprint	
URL – security-service/api/v2a/public/accounts/fingerprint/{fingerprint}	
Query	
{	
fingerprint (string)	Отпечаток сертификата
}	
Request	
–	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID учетной записи
login (string),	Логин учетной записи
principalName (string),	Отображаемое учетной записи
role (enum: ADMINISTRATOR, OPERATOR, UNKNOWN),	Роль учетной записи
status (enum: ACTIVE, BLOCKED, UNKNOWN),	Статус учетной записи
created (instant),	Время создания (ISO 8601)
isLinked (boolean)	Флаг: связь с субъектом из ресурсной системы
}	

МЕТОДЫ РАБОТЫ С ИЗДАТЕЛЯМИ

1.57 Метод поиска издателей

GET API – Поиск учетных записей	
Метод доступен администратору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#[v2a] Контроллер%3A Издатели/findAll_8	
URL – settings-service/api/v2a/public/issuer	
Query	
{	
authorityKeyIdentifier (string),	Идентификатор ключа издателя
subjectKeyIdentifier (string),	Идентификатор ключа сертификата ЦС
isActive (boolean),	Флаг: активный ЦС
sortDirection (string) [опционально],	Направления сортировки (ASC;DESC)
sortBy (string[]) [опционально],	Список полей, к которым применяется сортировка
pageOffset (integer) [опционально],	Смещение от начала списка (пагинация)
pageLimit (integer) [опционально]	Ограничение на размер выборки (пагинация)
}	
Request	
–	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID издателя
caId (UUID),	ID ЦС
validFrom (instant),	Дата начала действия сертификата ЦС (ISO 8601)
validTo (instant),	Дата окончания действия сертификата ЦС (ISO 8601)
name (string),	Имя сертификата ЦС (на основе CN)
title (string),	Отображаемое имя ЦС
subjectKeyIdentifier (string),	Идентификатор ключа сертификата ЦС
authorityKeyIdentifier (string),	Идентификатор ключа издателя сертификата ЦС
active (boolean)	Флаг: активный ЦС
}	

МЕТОДЫ РАБОТЫ С ПОДРАЗДЕЛЕНИЯМИ И ГРУППАМИ БЕЗОПАСНОСТИ

1.58 Метод поиска подразделений

GET API – Поиск подразделений	
Метод доступен администратору и оператору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A%20подразделения/findAll_5	
URL – subjects-service/api/v2a/public/organizational-units	
Query	
{	
sortDirection (string) [опционально],	Направления сортировки (ASC;DESC)
sortBy (string[]) [опционально],	Список полей, к которым применяется сортировка
pageOffset (integer) [опционально],	Смещение от начала списка (пагинация)
pageLimit (integer) [опционально],	Ограничение на размер выборки (пагинация)
id (UUID[]) [опционально],	ID подразделения
notId (UUID[]) [опционально],	Исключая ID подразделения
resourceId (UUID) [опционально],	ID ресурсной системы
subjectId (uuid) [опционально],	Идентификатор субъекта
search (string) [опционально]	Фильтр: полнотекстовый поиск
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID подразделения
commonName (string),	Имя подразделения
distinguishedName (string),	DN подразделения
resource: {	Ресурсная система
id (UUID),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string),	DN ресурсной системы
},	
modify (instant),	Дата и время обновления в ресурсной системе (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.59 Метод получения подразделения по идентификатору

GET API – Получение подразделения по идентификатору	
Метод доступен администратору и оператору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A%20подразделения/findById_5	
URL – subjects-service/api/v2a/public/organizational-units/{id}	
Query	
{	
id (uuid) [опционально]	ID подразделения
}	
Request	
-	
Response	
{	Ответ JSON в HTTP-body
id (UUID),	ID подразделения

commonName (string),	Имя подразделения
distinguishedName (string),	DN подразделения
resource: {	Ресурсная система
id (UUID),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string),	DN ресурсной системы
},	
modify (instant),	Дата и время обновления в ресурсной системе (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.60 Метод групп безопасности

GET API – Поиск групп безопасности	
Метод доступен администратору и оператору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A группы безопасности/findAll_3	
URL – subjects-service/api/v2a/public/security-groups	
Query	
{	
sortDirection (string) [опционально],	Направления сортировки (ASC;DESC)
sortBy (string[]) [опционально],	Список полей, к которым применяется сортировка
pageOffset (integer) [опционально],	Смещение от начала списка (пагинация)
pageLimit (integer) [опционально],	Ограничение на размер выборки (пагинация)
id (UUID[]) [опционально],	ID группы безопасности
notId (UUID[]) [опционально],	Исключая ID группы безопасности
resourceId (UUID) [опционально],	ID ресурсной системы
subjectId (uuid) [опционально],	Идентификатор субъекта
search (string) [опционально],	Фильтр: полнотекстовый поиск
isActive (boolean)	Флаг: активная для назначения прав группа безопасности
}	
Request	
-	
Response	
ResponseEntity -> CollectionResponse -> {	Ответ JSON в HTTP-body
id (UUID),	ID группы безопасности
commonName (string),	Имя группы безопасности
distinguishedName (string),	DN группы безопасности
resource: {	Ресурсная система
id (UUID),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string),	DN ресурсной системы
},	
isActive (boolean),	Флаг: активная для назначения прав группа безопасности
modify (instant),	Дата и время обновления в ресурсной системе (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

1.61 Метод получения группы безопасности по идентификатору

GET API – Получение группы безопасности по идентификатору	
Метод доступен администратору и оператору	
Swagger: https://HOST/external-integration-service/swagger/swagger-ui/index.html#/v2a/Контроллер%3A группы безопасности/findById_3	

URL – subjects-service/api/v2a/public/security-groups/{id}	
Query	
{	
id (uuid) [опционально]	ID группы безопасности
}	
Request	
–	
Response	
{	Ответ JSON в HTTP-body
id (UUID),	ID группы безопасности
commonName (string),	Имя группы безопасности
distinguishedName (string),	DN группы безопасности
resource: {	Ресурсная система
id (UUID),	ID ресурсной системы
commonName (string),	Имя ресурсной системы
distinguishedName (string),	DN ресурсной системы
},	
isActive (boolean),	Флаг: активная для назначения прав группа безопасности
modify (instant),	Дата и время обновления в ресурсной системе (ISO 8601)
updated (instant),	Время обновления (ISO 8601)
created (instant)	Время создания (ISO 8601)
}	

ДОПОЛНИТЕЛЬНЫЕ ВОЗМОЖНОСТИ

1.62 Выполнение администратором методов от имени другой учетной записи

Все методы за исключением метода идентификации и аутентификации по сертификату доступа (раздел 1.1) и метода получения версий сервисов (раздел 0) позволяют администратору использовать их от имени любой учетной записи Центра сертификации Aladdin eCA. Для этого необходимо передавать идентификатор учетной записи в заголовке `x-User-Context` HTTP-запроса при вызове методов.

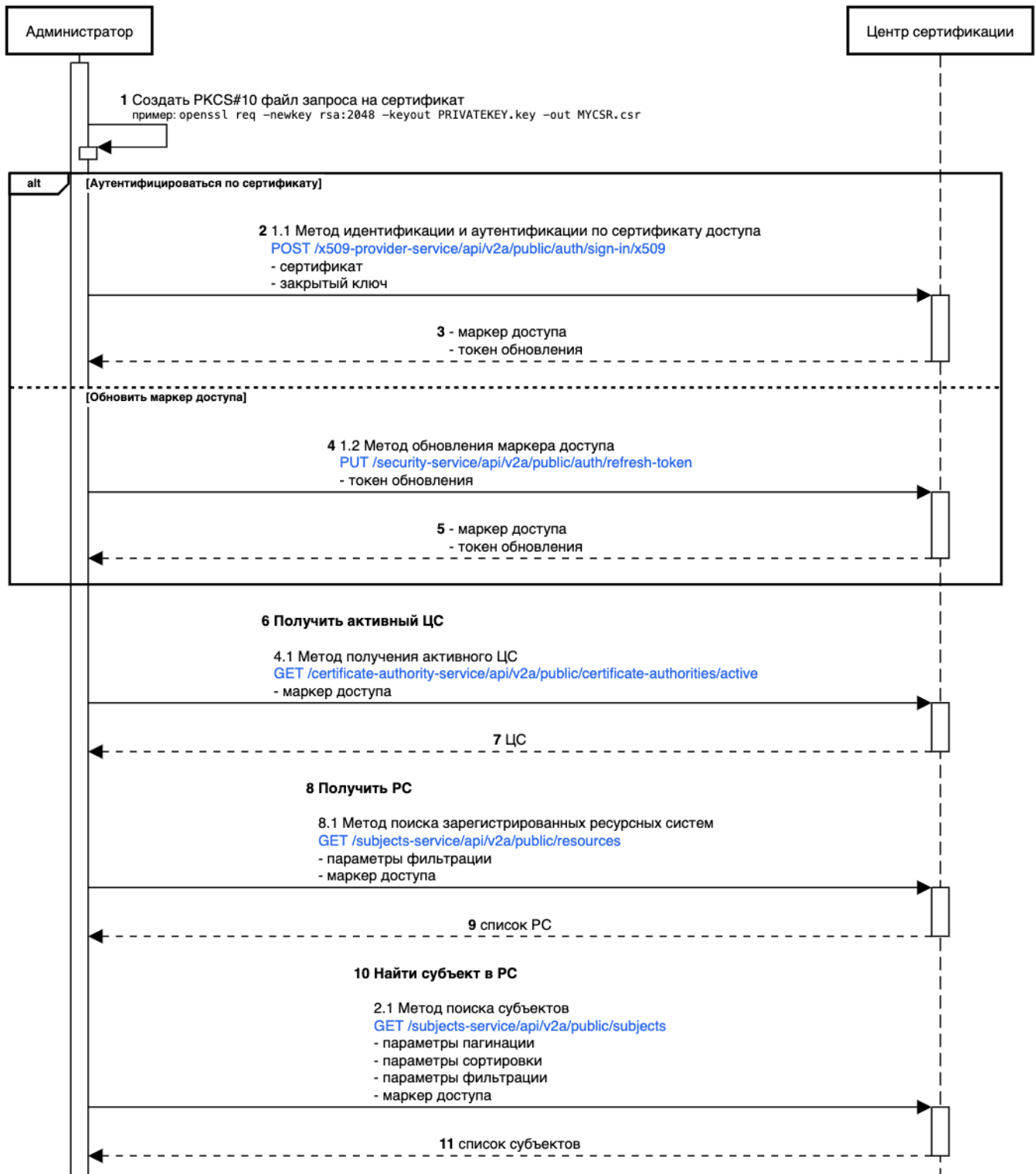
СОПОСТАВЛЕНИЕ МЕТОДОВ ОБРАТНОЙ СОВМЕСТИМОСТИ И ПУБЛИЧНОГО REST API

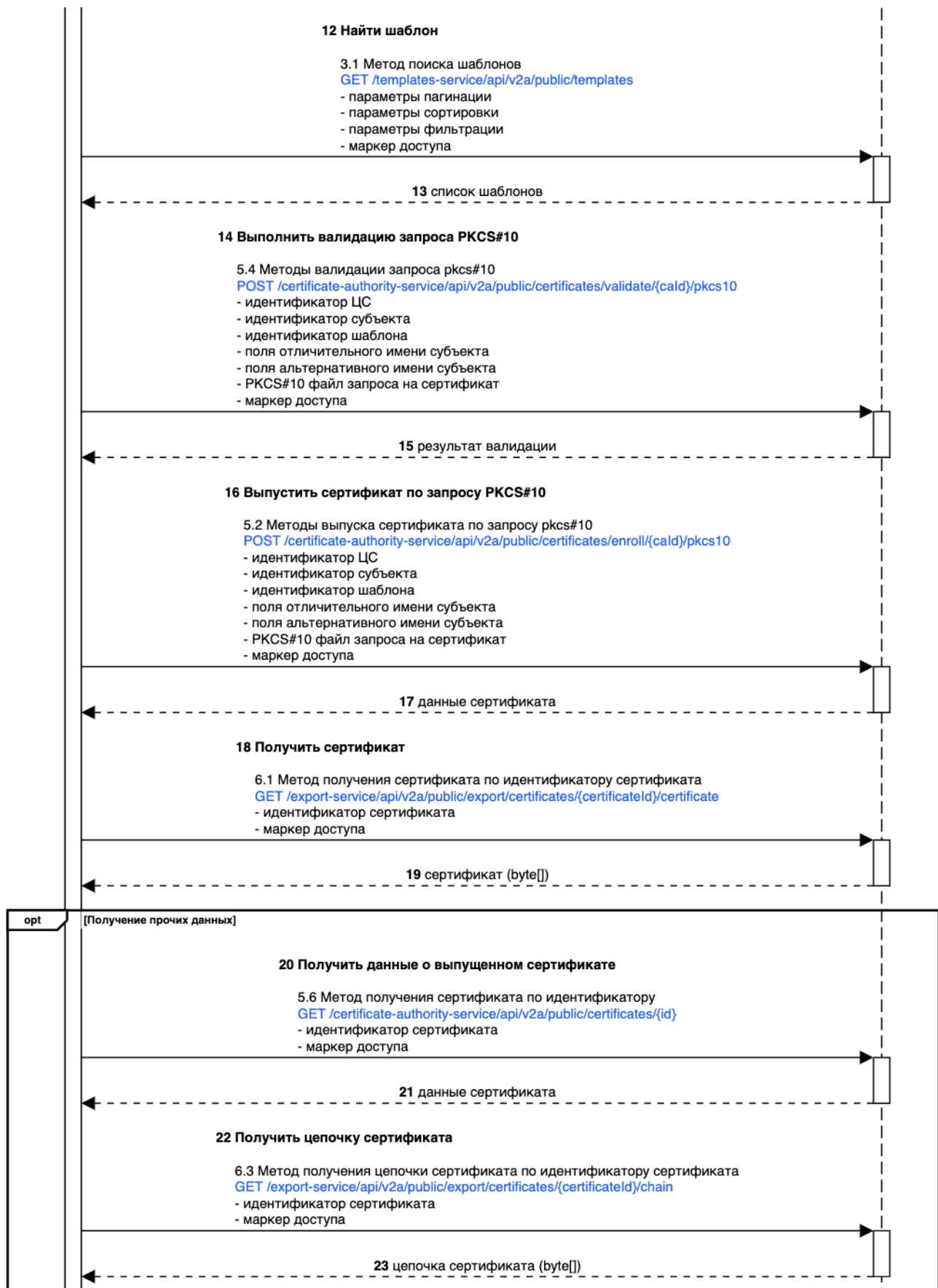
REST API обратной совместимости	Публичный REST API
GetCertificateTemplates (backward-compatibility-service/api/v2a/ui/certificates/certificatetemplates/dtos)	3.1 Метод поиска шаблонов
CreateAndEnrollCertificate (backward-compatibility-service/api/v2a/ui/certificates/create-and-enroll)	Реализуется следующими методами: 1) Создание субъекта или изменение его атрибутов – 2.4 Методы создания и изменения субъекта на основании запроса pkcs#10 2) Создание сертификата для субъекта – 5.2 Методы выпуска сертификата по запросу pkcs#10
ReactivateCertificate (backward-compatibility-service/api/v2a/ui/certificates/reactivate)	5.9 Метод активации сертификата по идентификатору
SuspendCertificate (backward-compatibility-service/api/v2a/ui/certificates/suspend)	5.8 Метод отзыва (приостановки) сертификата по идентификатору
RevokeCertificate (backward-compatibility-service/api/v2a/ui/certificates/revoke)	5.8 Метод отзыва (приостановки) сертификата по идентификатору
GetCertificateStatusByCertSerialNumber (backward-compatibility-service/api/v2a/ui/certificates/status/{certSerialNumber})	5.6 Метод получения сертификата по серийному номеру
PublishCrl (backward-compatibility-service/api/v2a/ui/cdp/publish-crl)	7.1 Метод генерации и публикации CRL по идентификатору Центра сертификации
GetEndEntityProfileAttributesByTemplateProfileId (backward-compatibility-service/api/v2a/ui/certificates/certificatetemplates/attributes/{id})	3.2 Метод получения шаблона по идентификатору

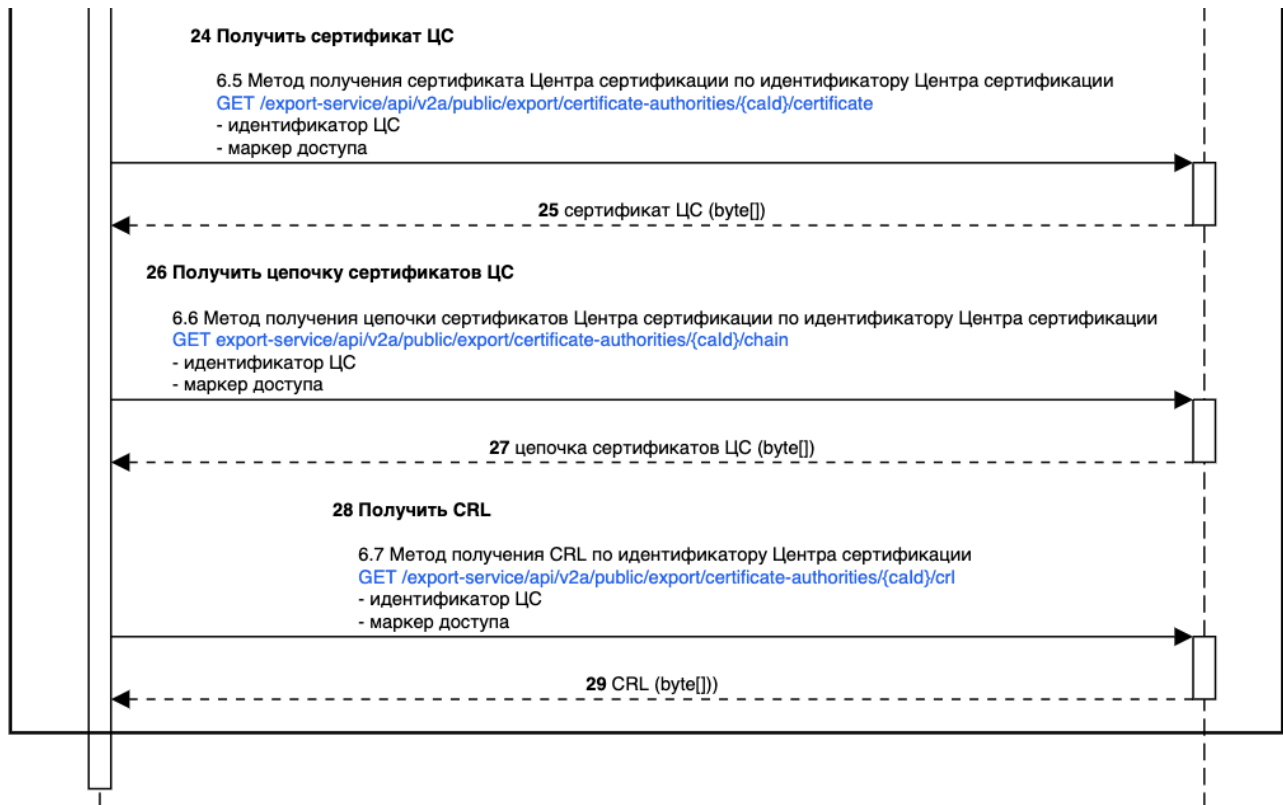
ДИАГРАММА ПОСЛЕДОВАТЕЛЬНОСТИ ПОЛУЧЕНИЯ СЕРТИФИКАТА ПО ЗАПРОСУ PKCS#10

1.63 Диаграмма последовательности получения сертификата по запросу PKCS#10

Центр сертификации
Выпуск сертификата по запросу PKCS#10







Для получения сертификата по запросу PKCS#10 следует выполнить шаги, представленные на диаграмме последовательности выше. Краткое описание представлено ниже:

- предварительно подготовить файл запроса (шаг 1) – `request`;
- аутентифицироваться на ЦС (шаг 2-3);
- при истечении маркера доступа следует произвести его обновление, используя токен обновления (шаги 4-5);
- все последующие запросы используют маркер доступа;
- после этого следует получить активный ЦС (шаги 6-7), далее будет использован его идентификатор – `caId`;
- получить идентификатор ресурсной системы, в которой находит субъект, для которого выпускается сертификат (шаги 8-9) – `resourceId`;
- найти субъект, для которого выпускается сертификат (шаги 10-11), при этом для поиска используется идентификатор ресурсной системы `resourceId` – получить его идентификатор `subjectId`;
- найти шаблон сертификата (шаги 12-13) – получить его идентификатор `templateId`;
- выполнить валидацию запроса (шаги 14-15), используются параметры:
 - идентификатор ЦС – `caId`;
 - идентификатор субъекта – `subjectId`;
 - идентификатор шаблона – `templateId`;
 - файл запроса PKCS#10 – `request`;
 - поля отличного имени субъекта;
 - поля альтернативного имени субъекта.

- при отсутствии ошибок выпустить сертификат по запросу PKCS#10 (шаги 16-17), параметры такие же, как при валидации запроса, в результате будет идентификатор сертификата – `certificateId`;
- после этого скачать выпущенный сертификат (шаги 18-19), используя `certificateId`;
- опционально можно получить информацию о выпущенном сертификате, цепочку сертификатов, используя идентификатор сертификата – `certificateId` (шаги 20-23).
- а также сертификат ЦС, цепочку сертификатов ЦС и CRL, используя идентификатор ЦС – `caId` (шаги 24-29).

СПЕЦИФИКАЦИЯ ОРЕНАРИ

```
{
  "openapi": "3.0.1",
  "info": {
    "title": "Aeca external integration Service API",
    "description": "API to external integration",
    "termsOfService": "http://example.com/terms/",
    "contact": {
      "name": "Product Service API Team",
      "url": "http://example.com/team",
      "email": "apiteam@example.com"
    },
    "license": {
      "name": "Apache 2.0",
      "url": "http://www.apache.org/licenses/LICENSE-2.0.html"
    },
    "version": "2.0.0",
    "servers": [
      {
        "url": "",
        "description": "Service URL"
      }
    ],
    "security": [
      {
        "X-API-KEY": []
      },
      {
        "token": []
      },
      {
        "X-AECA-User-ID": [],
        "X-AECA-User-Login": [],
        "X-AECA-User-Principal": [],
        "X-AECA-User-Role": [],
        "X-AECA-User-Authorities": [],
        "X-AECA-User-Organizational-Unit-Rights": [],
        "X-AECA-User-Security-Group-Rights": [],
        "X-AECA-User-Resource-Rights": [],
        "X-User-Session-Id": []
      }
    ],
    "tags": [
      {
        "name": "[v2a] Контроллер: Точки подключения ресурсных систем",
        "description": "REST API: Точки подключения ресурсных систем"
      },
      {
        "name": "[v2a] Контроллер: Учетные записи",
        "description": "REST API: Учетные записи"
      },
      {
        "name": "version-controller",
        "description": "API методов: Получение версии модуля"
      },
      {
        "name": "[v2] Контроллер: сертификаты",
        "description": "API методов: сертификаты"
      },
      {
        "name": "[v2] Контроллер: ресурсные системы",
        "description": "REST API: ресурсные системы"
      },
      {
        "name": "[v2a] Контроллер: шаблоны",
        "description": "REST API: шаблоны"
      },
      {
        "name": "[v2] Контроллер: Точки подключения ресурсных систем",
        "description": "REST API: Точки подключения ресурсных систем"
      },
      {
        "name": "[v2a] Контроллер экспорта",
        "description": "API методов по работе с экспорта"
      },
      {
        "name": "[v2] Контроллер: Центры сертификации",
        "description": "API методов: Центры сертификации"
      },
      {
        "name": "[v2a] Контроллер: ресурсные системы",
        "description": "REST API: ресурсные системы"
      },
      {
        "name": "[v2a] Контроллер: сертификаты",
        "description": "API методов: сертификаты"
      },
      {
        "name": "[v2] Контроллер: субъекты",
        "description": "REST API: субъекты"
      },
      {
        "name": "[v2] Контроллер: Авторизации",
        "description": "Интерфейс описания Api: Авторизации"
      },
      {
        "name": "[v2a] Контроллер: контейнеры сертификатов",
        "description": "API методов: контейнеры сертификатов"
      },
      {
        "name": "[v2a] Контроллер: Издатели",
        "description": "REST API: Издатели"
      },
      {
        "name": "[v2a] Контроллер: подразделения",
        "description": "REST API: подразделения"
      },
      {
        "name": "[v2a] Контроллер: группы безопасности",
        "description": "REST API: группы безопасности"
      },
      {
        "name": "[v2] Контроллер: шаблоны",
        "description": "REST API: шаблоны"
      },
      {
        "name": "[v2] Контроллер: Конфигурации CRL для ЦС",
        "description": "API методов: Конфигурации CRL для ЦС"
      },
      {
        "name": "[v2] Контроллер: Syslog сервера",
        "description": "REST API: Syslog сервера"
      },
      {
        "name": "[v2a] Контроллер: Авторизации",
        "description": "Интерфейс описания Api: Авторизации"
      },
      {
        "name": "[v2a] Контроллер: Syslog сервера",
        "description": "REST API: Syslog сервера"
      },
      {
        "name": "[v2a] Контроллер: Конфигурации CRL для ЦС",
        "description": "API методов: Конфигурации CRL для ЦС"
      },
      {
        "name": "[v2a] Контроллер: субъекты",
        "description": "REST API: субъекты"
      },
      {
        "name": "[v2a] Контроллер: Центры сертификации",
        "description": "API методов: Центры сертификации"
      },
      {
        "name": "[v2] Контроллер экспорта",
        "description": "API методов по работе с экспорта"
      }
    ],
    "paths": {
      "/subjects-service/api/v2a/public/subjects": {
        "get": {
          "tags": [
            "[v2a] Контроллер: субъекты"
          ],
          "summary": "Поиск субъектов",
          "description": "GET метод REST API: поиск субъектов",
          "operationId": "findAll_2",
          "parameters": [
            {
              "name": "id",
              "in": "query",
              "description": "Фильтр: ID субъекта",
              "schema": {
                "type": "array",
                "items": {
                  "type": "string",
                  "format": "uuid"
                }
              },
              "name": "notId",
              "in": "query",
              "description": "Фильтр: исключая ID субъекта",
              "schema": {
                "type": "array",
                "items": {
                  "type": "string",
                  "format": "uuid"
                }
              },
              "name": "resourceId",
              "in": "query",
              "description": "Фильтр: ID ресурсной системы",
              "schema": {
                "type": "array",
                "items": {
                  "type": "string",
                  "format": "uuid"
                }
              },
              "name": "securityGroupId",
              "in": "query",
              "description": "Фильтр: ID группы безопасности",
              "schema": {
                "type": "array",
                "items": {
                  "type": "string",
                  "format": "uuid"
                }
              },
              "name": "organizationalUnitId",
              "in": "query",
              "description": "Фильтр: ID подразделения",
              "schema": {
                "type": "array",
                "items": {
                  "type": "string",
                  "format": "uuid"
                }
              },
              "name": "search",
              "in": "query",
              "description": "Фильтр: полнотекстовый поиск по имени",
              "schema": {
                "type": "string",
                "default": "Name"
              },
              "name": "isConnected",
              "in": "query",
              "description": "Фильтр: подключение субъекта к ресурсной системе",
              "schema": {
                "type": "boolean"
              },
              "name": "isBlocked",
              "in": "query",
              "description": "Фильтр: блокировка субъекта в ресурсной системе",
              "schema": {
                "type": "boolean"
              },
              "name": "pageLimit",
              "in": "query",
              "description": "Пагинация: ограничение на размер списка",
              "schema": {
                "minimum": 0,
                "type": "integer",
                "default": 10000
              },
              "name": "pageOffset",
              "in": "query",
              "description": "Пагинация: смещение от начала"
            }
          ]
        }
      }
    }
  }
}
```

```

списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "description": "Сортировка: имя поля, по которому выполнять сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortDirection", "in": "query", "description": "Сортировка: направление сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN", "asc", "desc"]}}, {"responses": {"200": {"description": "Список найденных субъектов", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseSubjectV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"put": {"tags": ["v2a"], "summary": "Обновление (создание) субъекта", "description": "GET метод REST API: Обновление (создание) субъекта", "operationId": "update", "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/EditSubjectV2aRequestPublic"}}}}, {"required": true, "responses": {"200": {"description": "Субъект успешно обновлен (создан)", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseSubjectV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"subjects-service/api/v2a/public/subjects/pkcs10": {"put": {"tags": ["v2a"], "summary": "Обновление (создание) субъекта на основании запроса #PKCS10", "description": "GET метод REST API: Обновление (создание) субъекта на основании запроса #PKCS10", "operationId": "updateByPkcs10AsMultipartFile_1", "requestBody": {"content": {"multipart/form-data": {"schema": {"$ref": "#/components/schemas/EditSubjectByCsrV2aRequestPublicMultipartFile"}}}}, {"application/json": {"schema": {"$ref": "#/components/schemas/EditSubjectByCsrV2aRequestPublicFileDescription"}}}}, {"required": true, "responses": {"200": {"description": "Субъект успешно обновлен (создан)", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseSubjectV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"oneOf": [{"$ref": "#/components/schemas/ItemResponseSubjectV2aResponsePublic"}, {"$ref": "#/components/schemas/ErrorResponse"}]}}}}}}, {"subjects-service/api/v2/public/subjects": {"get": {"tags": ["v2"], "summary": "Поиск субъектов", "description": "GET метод REST API: поиск субъектов", "operationId": "findAll_6", "parameters": [{"name": "id", "in": "query", "description": "Фильтр: ID субъекта", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "resourceId", "in": "query", "description": "Фильтр: ID ресурсной системы", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "securityGroupId", "in": "query", "description": "Фильтр: ID группы безопасности", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "organizationalUnitId", "in": "query", "description": "Фильтр: ID подразделения", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "search", "in": "query", "description": "Фильтр: полнотекстовый поиск по имени", "schema": {"type": "string", "default": "Name"}}, {"name": "isConnected", "in": "query", "description": "Фильтр: подключение субъекта к ресурсной системе", "schema": {"type": "boolean"}}, {"name": "isBlocked", "in": "query", "description": "Фильтр: блокировка субъекта в ресурсной системе", "schema": {"type": "boolean"}}, {"name": "pageLimit", "in": "query", "description": "Пагинация: ограничение на размер списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "query", "description": "Пагинация: смещение от начала списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "description": "Сортировка: имя поля, по которому выполнять сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortDirection", "in": "query", "description": "Сортировка: направление сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN", "asc", "desc"]}}, {"responses": {"200": {"description": "Список найденных субъектов", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseSubjectResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"put": {"tags": ["v2"], "summary": "Обновление (создание) субъекта", "description": "GET метод REST API: Обновление (создание) субъекта"}

```

```

субъекта", "operationId": "update_1", "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/EditSubjectRequestPublic"}}}, "required": true}, "responses": {"200": {"description": "Субъект успешно обновлен (создан)", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ItemResponseSubjectResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/subjects-service/api/v2/public/subjects/pkcs10": {"put": {"tags": ["v2] Контроллер: субъекты"], "summary": "Обновление (создание) субъекта на основании запроса #PKCS10", "description": "GET метод REST API: Обновление (создание) субъекта на основании запроса #PKCS10", "operationId": "updateByPkcs10AsMultipartFile_2", "requestBody": {"content": {"multipart/form-data": {"schema": {"$ref": "#/components/schemas/EditSubjectByCsrRequestPublicMultipartFile"}}}, "application/json": {"schema": {"$ref": "#/components/schemas/EditSubjectByCsrRequestPublicFileDescription"}}}, "required": true}, "responses": {"200": {"description": "Субъект успешно обновлен (создан)", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ItemResponseSubjectResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*/*": {"schema": {"oneOf": [{"$ref": "#/components/schemas/ItemResponseSubjectResponsePublic"}, {"$ref": "#/components/schemas/ErrorResponse"}]}}}}}}, "/logs-service/api/v2a/public/syslog/{id}": {"get": {"tags": ["v2a] Контроллер: Syslog сервера"], "summary": "Получение syslog сервера по ID", "description": "GET метод REST API получения syslog сервера по ID", "operationId": "findById_9", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор syslog сервера типа UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Syslog сервер успешно получено", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ItemResponseSyslogV2aResponsePublic"}}}}, "404": {"description": "Syslog сервер отсутствует в журнале", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"put": {"tags": ["v2a] Контроллер: Syslog сервера"], "summary": "Обновление syslog сервера по ID", "description": "POST метод REST API обновление syslog сервера по ID", "operationId": "updateById", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор syslog сервера типа UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/CreateSyslogV2aRequestPublic"}}}, "required": true}, "responses": {"200": {"description": "Syslog сервера успешно отредактирован", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ItemResponseSyslogV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"delete": {"tags": ["v2a] Контроллер: Syslog сервера"], "summary": "Удаление syslog сервера по ID", "description": "DELETE метод REST API: Удаление syslog сервера по ID", "operationId": "deleteById_2", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор syslog сервера типа UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Syslog сервер удален"}, "404": {"description": "syslog сервер не найден", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}}}, {"get": {"tags": ["v2] Контроллер: Syslog сервера"], "summary": "Получение syslog сервера по ID", "description": "GET метод REST API получения syslog сервера по ID", "operationId": "findById_10", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор syslog сервера типа UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Syslog сервер успешно получено", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ItemResponseSyslogResponsePublic"}}}}, "404": {"description": "Syslog сервер отсутствует в журнале", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя

```

```

ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "put": {"
tags": ["[v2] Контроллер: Syslog сервера"], "summary": "Обновление syslog сервера по
ID", "description": "POST метод REST API обновление syslog сервера по
ID", "operationId": "updateById_1", "parameters": [{"name": "id", "in": "path", "description": "Идентиф
икатор syslog сервера типа
UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "requestBody": {"content": {"
application/json": {"schema": {"$ref": "#/components/schemas/CreateSyslogRequestPublic"}}}, "requi
red": true}, "responses": {"200": {"description": "Syslog сервера успешно
отредактирован", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ItemResp
onseSyslogResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}
}}}}, "delete": {"tags": ["[v2] Контроллер: Syslog сервера"], "summary": "Удаление syslog сервера
по ID", "description": "DELETE метод REST API: Удаление syslog сервера по
ID", "operationId": "deleteById_3", "parameters": [{"name": "id", "in": "path", "description": "Идентиф
икатор syslog сервера типа
UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"descri
ption": "Syslog сервер удален"}, "404": {"description": "syslog сервер не
найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/ldap-
service/api/v2a/public/resources/{id}/synchronize": {"put": {"tags": ["[v2a] Контроллер:
ресурсные системы"], "summary": "Синхронизация ресурсной системы по ID", "description": "PUT метод
REST API: синхронизация ресурсной системы по
ID", "operationId": "synchronize", "parameters": [{"name": "id", "in": "path", "description": "ID
ресурсной
системы", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"des
cription": "Процесс синхронизации запущен"}, "404": {"description": "Ресурсная система с указанным
ID не
существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500":
{"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/ldap-
service/api/v2a/public/connection-points/{id}/synchronize": {"put": {"tags": ["[v2a] Контроллер:
Точки подключения ресурсных систем"], "summary": "Синхронизация точки подключения по
ID", "description": "PUT метод REST API: синхронизация точки подключения по
ID", "operationId": "synchronize_1", "parameters": [{"name": "id", "in": "path", "description": "ID
точки
подключения", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {
"description": "Процесс синхронизации запущен"}, "404": {"description": "Точка подключения с
указанным ID не
существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500":
{"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/ldap-
service/api/v2/public/resources/{id}/synchronize": {"put": {"tags": ["[v2] Контроллер: ресурсные
системы"], "summary": "Синхронизация ресурсной системы по ID", "description": "PUT метод REST API:
синхронизация ресурсной системы по
ID", "operationId": "synchronize_2", "parameters": [{"name": "id", "in": "path", "description": "ID
ресурсной
системы", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"des
cription": "Процесс синхронизации запущен"}, "404": {"description": "Ресурсная система с указанным
ID не
существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500":
{"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/ldap-
service/api/v2/public/connection-points/{id}/synchronize": {"put": {"tags": ["[v2] Контроллер:
Точки подключения ресурсных систем"], "summary": "Синхронизация точки подключения по
ID", "description": "PUT метод REST API: синхронизация точки подключения по
ID", "operationId": "synchronize_3", "parameters": [{"name": "id", "in": "path", "description": "ID
точки
подключения", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {
"description": "Процесс синхронизации запущен"}, "404": {"description": "Точка подключения с
указанным ID не

```

```

существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500":
{"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2a/public/certificates/{id}/revoke": {"put": {"tags": ["v2a]
Контроллер: сертификаты"], "summary": "Метод отзыва сертификата по его
идентификатору", "description": "PUT метод REST API: Метод отзыва
сертификата", "operationId": "revokeById", "parameters": [{"name": "id", "in": "path", "description": "
Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "requestBody": {"cont
ent": {"application/json": {"schema": {"$ref": "#/components/schemas/RevokeCertificateV2aRequestPu
blic"}}}, "required": true}, "responses": {"200": {"description": "Сертификат
отозван"}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2a/public/certificates/{id}/reactivate": {"put": {"tags": ["v2a]
Контроллер: сертификаты"], "summary": "Метод активации сертификата по его
идентификатору", "description": "PUT метод REST API: Метод активации
сертификата", "operationId": "reactivateById", "parameters": [{"name": "id", "in": "path", "descriptio
n": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {
"description": "Сертификат активирован"}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2a/public/certificates/{id}/publish": {"put": {"tags": ["v2a]
Контроллер: сертификаты"], "summary": "Публикация в ресурсную систему сертификата по
id", "description": "GET метод REST API: публикация в ресурсную систему сертификата по
id", "operationId": "publishById", "parameters": [{"name": "id", "in": "path", "description": "Идентифи
катор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {
"description": "Сертификат опубликован"}, "404": {"description": "Сертификат не
найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2/public/certificates/{id}/revoke": {"put": {"tags": ["v2]
Контроллер: сертификаты"], "summary": "Метод отзыва сертификата по его
идентификатору", "description": "PUT метод REST API: Метод отзыва
сертификата", "operationId": "revokeById_1", "parameters": [{"name": "id", "in": "path", "description"
: "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "requestBody": {"cont
ent": {"application/json": {"schema": {"$ref": "#/components/schemas/RevokeCertificateRequestPubli
c"}}}, "required": true}, "responses": {"200": {"description": "Сертификат
отозван"}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2/public/certificates/{id}/reactivate": {"put": {"tags": ["v2]
Контроллер: сертификаты"], "summary": "Метод активации сертификата по его
идентификатору", "description": "PUT метод REST API: Метод активации
сертификата", "operationId": "reactivateById_1", "parameters": [{"name": "id", "in": "path", "descript
ion": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {
"description": "Сертификат активирован"}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2/public/certificates/{id}/publish": {"put": {"tags": ["v2]
Контроллер: сертификаты"], "summary": "Публикация в ресурсную систему сертификата по
id", "description": "GET метод REST API: публикация в ресурсную систему сертификата по
id", "operationId": "publishById_1", "parameters": [{"name": "id", "in": "path", "description": "Иденти
фикатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {
"description": "Сертификат опубликован"}, "404": {"description": "Сертификат не
найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/x509-
provider-service/api/v2a/public/auth/sign-in/x509": {"post": {"tags": ["v2a] Контроллер:
Авторизации"], "summary": "Аутентификация x509", "description": "POST метод REST API:

```

```

Аутентификация x509", "operationId": "signInByX509", "responses": {"200": {"description": "Данные о токене", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseTokenV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/x509-provider-service/api/v2/public/auth/sign-in/x509": {"post": {"tags": ["v2] Контроллер: Авторизации"], "summary": "Аутентификация x509", "description": "POST метод REST API: Аутентификация x509", "operationId": "signInByX509_1", "responses": {"200": {"description": "Данные о токене", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseTokenResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/security-service/api/v2a/public/auth/refresh-token": {"post": {"tags": ["v2a] Контроллер: Авторизации"], "summary": "Обновление токена", "description": "POST метод REST API: Обновление токена", "operationId": "refreshToken", "responses": {"200": {"description": "Данные о токене", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseTokenV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/security-service/api/v2/public/auth/refresh-token": {"post": {"tags": ["v2] Контроллер: Авторизации"], "summary": "Обновление токена", "description": "POST метод REST API: Обновление токена", "operationId": "refreshToken_1", "responses": {"200": {"description": "Данные о токене", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseTokenResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/publisher-service/api/v2a/ui/certificate-authorities/{caId}/crl-configuration/generate": {"post": {"tags": ["v2a] Контроллер: Конфигурации CRL для ЦС"], "summary": "Генерация CRL по ID ЦС (Deprecated)", "description": "POST метод REST API: Генерация CRL по ID ЦС (Deprecated)", "operationId": "generateDeprecated", "parameters": [{"name": "caId", "in": "path", "description": "Идентификатор ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Успешная публикация"}, "404": {"description": "ЦС не найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/publisher-service/api/v2a/public/certificate-authorities/{caId}/crl-configuration/generate": {"post": {"tags": ["v2a] Контроллер: Конфигурации CRL для ЦС"], "summary": "Генерация CRL по ID ЦС", "description": "POST метод REST API: Генерация CRL по ID ЦС", "operationId": "generate", "parameters": [{"name": "caId", "in": "path", "description": "Идентификатор ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Успешная публикация"}, "404": {"description": "ЦС не найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/publisher-service/api/v2/ui/certificate-authorities/{caId}/crl-configuration/generate": {"post": {"tags": ["v2] Контроллер: Конфигурации CRL для ЦС"], "summary": "Генерация CRL по ID ЦС (Deprecated)", "description": "POST метод REST API: Генерация CRL по ID ЦС (Deprecated)", "operationId": "generateDeprecated_1", "parameters": [{"name": "caId", "in": "path", "description": "Идентификатор ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Успешная публикация"}, "404": {"description": "ЦС не найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/publisher-service/api/v2/public/certificate-authorities/{caId}/crl-configuration/generate": {"post": {"tags": ["v2] Контроллер: Конфигурации CRL для ЦС"], "summary": "Генерация CRL по ID ЦС", "description": "POST метод REST API: Генерация CRL по ID ЦС", "operationId": "generate_1", "parameters": [{"name": "caId", "in": "path", "description": "Идентиф

```

икатор

```

"required":true,"schema":{"type":"string","format":"uuid"}}},"responses":{"204":{"description":"Успешная публикация"},"404":{"description":"ЦС не найден","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}},"500":{"description":"Внутренняя ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"/logs-service/api/v2a/public/syslog":{"get":{"tags":["[v2a] Контроллер: Syslog сервера"],"summary":"Поиск syslog сервера","description":"GET метод REST API поиск syslog сервера","operationId":"findAll_10","responses":{"200":{"description":"Syslog сервера получены","content":{"*//*":{"schema":{"$ref":"#/components/schemas/CollectionResponseSyslogV2aResponsePublic"}}}}},"500":{"description":"Внутренняя ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"post":{"tags":["[v2a] Контроллер: Syslog сервера"],"summary":"Создание syslog сервера","description":"POST метод REST API создание syslog сервера","operationId":"create","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/CreateSyslogV2aRequestPublic"}}},"required":true},"responses":{"201":{"description":"Syslog сервера успешно сохранен","content":{"application/json":{"schema":{"$ref":"#/components/schemas/ItemResponseSyslogV2aResponsePublic"}}}}},"500":{"description":"Внутренняя ошибка","content":{"application/json":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"/logs-service/api/v2/public/syslog":{"get":{"tags":["[v2] Контроллер: Syslog сервера"],"summary":"Поиск syslog сервера","description":"GET метод REST API поиск syslog сервера","operationId":"findAll_11","responses":{"200":{"description":"Syslog сервера получены","content":{"*//*":{"schema":{"$ref":"#/components/schemas/CollectionResponseSyslogResponsePublic"}}}}},"500":{"description":"Внутренняя ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"post":{"tags":["[v2] Контроллер: Syslog сервера"],"summary":"Создание syslog сервера","description":"POST метод REST API создание syslog сервера","operationId":"create_1","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/CreateSyslogRequestPublic"}}},"required":true},"responses":{"201":{"description":"Syslog сервера успешно сохранен","content":{"application/json":{"schema":{"$ref":"#/components/schemas/ItemResponseSyslogResponsePublic"}}}}},"500":{"description":"Внутренняя ошибка","content":{"application/json":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"/certificate-authority-service/api/v2a/public/parse/pkcs12":{"post":{"tags":["[v2a] Контроллер: контейнеры сертификатов"],"summary":"Расшифровка контейнера сертификата","description":"POST метод REST API: Расшифровка контейнера сертификата","operationId":"parse","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/ParseContainerV2aRequestPublic"}}},"required":true},"responses":{"200":{"description":"Данные контейнера сертификата","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ItemResponseContainerDetailsV2aResponsePublic"}}}}},"500":{"description":"Внутренняя ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"/certificate-authority-service/api/v2a/public/parse/pkcs10":{"post":{"tags":["[v2a] Контроллер: контейнеры сертификатов"],"summary":"Расшифровка контейнера сертификата","description":"POST метод REST API: Расшифровка сертификата","operationId":"parse_1","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/ParseCsrV2aRequestPublic"}}},"required":true},"responses":{"200":{"description":"Данные контейнера сертификата","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ItemResponseContainerDetailsPkcs10V2aResponsePublic"}}}}},"500":{"description":"Внутренняя ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"/certificate-authority-service/api/v2a/public/parse/pem":{"post":{"tags":["[v2a] Контроллер: контейнеры сертификатов"],"summary":"Расшифровка сертификата","description":"POST метод REST API: Расшифровка сертификата","operationId":"parse_2","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/ParsePemV2aRequestPublic"}}},"required":true},"responses":{"200":{"description":"Карточка сертификата","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ItemResponseCertificateV2aResponsePublic"}}}}},"500":{"description":"Внутренняя ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"/certi

```

```

ficate-authority-
service/api/v2a/public/certificates/validate/{caId}/pkcs10":{"post":{"tags":["v2a]
Контроллер: сертификаты"],"summary":"Валидация полей сертификата","description":"POST метод
REST API: Валидация полей
сертификата","operationId":"validate_1","parameters":[{"name":"caId","in":"path","description"
:"Идентификатор
ЦС","required":true,"schema":{"type":"string","format":"uuid"}},{"name":"accountId","in":"quer
y","description":"Идентификатор
аккаунта","required":false,"schema":{"type":"string","format":"uuid"}},{"name":"subjectId","in
":"query","description":"Идентификатор
субъекта","required":false,"schema":{"type":"string","format":"uuid"}}],"requestBody":{"conten
t":{"multipart/form-
data":{"schema":{"$ref":"#/components/schemas/EnrollCsrV2aRequestPublicMultipartFile"}}, "appli
cation/json":{"schema":{"$ref":"#/components/schemas/EnrollCsrV2aRequestPublicFileDescription"
}}},"required":true},"responses":{"200":{"description":"Данные
полей","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ItemResponseValidateV2aRespons
ePublic"}}},"500":{"description":"Внутренняя
ошибка","content":{"*//*":{"schema":{"oneOf":[{"$ref":"#/components/schemas/ItemResponseValidat
eV2aResponsePublic"}, {"$ref":"#/components/schemas/ErrorResponse"}]}}}}}},"/certificate-
authority-service/api/v2a/public/certificates/renewal/{caId}/pkcs10":{"post":{"tags":["v2a]
Контроллер: сертификаты"],"summary":"Перевыпуск сертификата с по запросу для
ЦС","description":"POST метод REST API: Перевыпуск сертификата по
запросу","operationId":"renewalRequestByCaId_1","parameters":[{"name":"caId","in":"path","desc
ription":"Идентификатор
ЦС","required":true,"schema":{"type":"string","format":"uuid"}},{"name":"userId","in":"query",
"description":"Идентификатор
юзера","required":false,"schema":{"type":"string","format":"uuid"}},{"name":"subjectId","in":"
query","description":"Идентификатор
субъекта","required":false,"schema":{"type":"string","format":"uuid"}}],"requestBody":{"conten
t":{"multipart/form-
data":{"schema":{"$ref":"#/components/schemas/EnrollCsrV2aRequestPublicMultipartFile"}}, "appli
cation/json":{"schema":{"$ref":"#/components/schemas/EnrollCsrV2aRequestPublicFileDescription"
}}},"required":true},"responses":{"200":{"description":"Карточка выпущенного
сертификата","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ItemResponseCreateCertif
icateV2aResponsePublic"}}},"500":{"description":"Внутренняя
ошибка","content":{"*//*":{"schema":{"oneOf":[{"$ref":"#/components/schemas/ItemResponseCreateC
ertificateV2aResponsePublic"}, {"$ref":"#/components/schemas/ErrorResponse"}]}}}}}},"/certific
ate-authority-service/api/v2a/public/certificates/enroll/{caId}":{"post":{"tags":["v2a]
Контроллер: сертификаты"],"summary":"Выпуск сертификата с закрытым ключом для
ЦС","description":"POST метод REST API: Выпуск сертификата с закрытым
ключом","operationId":"enrollByCaId","parameters":[{"name":"caId","in":"path","description":"И
дентификатор
ЦС","required":true,"schema":{"type":"string","format":"uuid"}},{"name":"userId","in":"query",
"description":"Идентификатор
юзера","required":false,"schema":{"type":"string","format":"uuid"}},{"name":"subjectId","in":"
query","description":"Идентификатор
субъекта","required":false,"schema":{"type":"string","format":"uuid"}}],"requestBody":{"conten
t":{"application/json":{"schema":{"$ref":"#/components/schemas/EnrollCertificateV2aRequestPubl
ic"}}},"required":true},"responses":{"200":{"description":"Карточка выпущенного
сертификата","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ItemResponseCreateCertif
icateV2aResponsePublic"}}},"500":{"description":"Внутренняя
ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}"},"/certi
ficate-authority-
service/api/v2a/public/certificates/enroll/{caId}/pkcs10":{"post":{"tags":["v2a] Контроллер:
сертификаты"],"summary":"Выпуск сертификата с по запросу для ЦС","description":"POST метод
REST API: Выпуск сертификата по
запросу","operationId":"enrollRequestByCaId_1","parameters":[{"name":"caId","in":"path","desc
ription":"Идентификатор
ЦС","required":true,"schema":{"type":"string","format":"uuid"}},{"name":"userId","in":"query",
"description":"Идентификатор
юзера","required":false,"schema":{"type":"string","format":"uuid"}},{"name":"subjectId","in":"

```

```

query", "description": "Идентификатор
субъекта", "required": false, "schema": {"type": "string", "format": "uuid"}}, "requestBody": {"content": {"multipart/form-
data": {"schema": {"$ref": "#/components/schemas/EnrollCsrV2aRequestPublicMultipartFile"}}, "application/json": {"schema": {"$ref": "#/components/schemas/EnrollCsrV2aRequestPublicFileDescription"}}, "required": true}, "responses": {"200": {"description": "Карточка выпущенного
сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseCreateCertificateV2aResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"oneOf": [{"$ref": "#/components/schemas/ItemResponseCreateCertificateV2aResponsePublic"}, {"$ref": "#/components/schemas/ErrorResponse"}]}}}}}, "/certificate-authority-
service/api/v2/public/certificates/validate/{caId}/pkcs10": {"post": {"tags": ["v2] Контроллер:
сертификаты"], "summary": "Валидация полей сертификата", "description": "POST метод REST API:
Валидация полей
сертификата", "operationId": "validate_2", "parameters": [{"name": "caId", "in": "path", "description":
"Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}, {"name": "accountId", "in": "query",
"description": "Идентификатор
аккаунта", "required": false, "schema": {"type": "string", "format": "uuid"}}, {"name": "subjectId", "in":
"query", "description": "Идентификатор
субъекта", "required": false, "schema": {"type": "string", "format": "uuid"}}}, "requestBody": {"content": {"multipart/form-
data": {"schema": {"$ref": "#/components/schemas/EnrollCsrRequestPublicMultipartFile"}}, "application/json": {"schema": {"$ref": "#/components/schemas/EnrollCsrRequestPublicFileDescription"}}, "required": true}, "responses": {"200": {"description": "Данные
полей", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseValidateResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"oneOf": [{"$ref": "#/components/schemas/ItemResponseValidateResponsePublic"}, {"$ref": "#/components/schemas/ErrorResponse"}]}}}}}, "/certificate-
authority-service/api/v2/public/certificates/enroll/{caId}": {"post": {"tags": ["v2] Контроллер:
сертификаты"], "summary": "Выпуск сертификата с закрытым ключом для ЦС", "description": "POST
метод REST API: Выпуск сертификата с закрытым
ключом", "operationId": "enrollByCaId_1", "parameters": [{"name": "caId", "in": "path", "description":
"Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}, {"name": "userId", "in": "query",
"description": "Идентификатор
юзера", "required": false, "schema": {"type": "string", "format": "uuid"}}, {"name": "subjectId", "in": "
query", "description": "Идентификатор
субъекта", "required": false, "schema": {"type": "string", "format": "uuid"}}}, "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/EnrollCertificateRequestPublic"}}, "required": true}, "responses": {"200": {"description": "Карточка выпущенного
сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseCreateCertificateResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/certificate-authority-
service/api/v2/public/certificates/enroll/{caId}/pkcs10": {"post": {"tags": ["v2] Контроллер:
сертификаты"], "summary": "Выпуск сертификата с по запросу для ЦС", "description": "POST метод
REST API: Выпуск сертификата по
запросу", "operationId": "enrollRequestByCaId_2", "parameters": [{"name": "caId", "in": "path", "description": "Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}, {"name": "userId", "in": "query",
"description": "Идентификатор
юзера", "required": false, "schema": {"type": "string", "format": "uuid"}}, {"name": "subjectId", "in": "
query", "description": "Идентификатор
субъекта", "required": false, "schema": {"type": "string", "format": "uuid"}}}, "requestBody": {"content": {"multipart/form-
data": {"schema": {"$ref": "#/components/schemas/EnrollCsrRequestPublicMultipartFile"}}, "application/json": {"schema": {"$ref": "#/components/schemas/EnrollCsrRequestPublicFileDescription"}}, "required": true}, "responses": {"200": {"description": "Карточка выпущенного
сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseCreateCertificateResponsePublic"}}}}, "500": {"description": "Внутренняя

```

```

ошибка", "content": {"*//*": {"schema": {"oneOf": [{"$ref": "#/components/schemas/ItemResponseCreateCertificateResponsePublic"}, {"$ref": "#/components/schemas/ErrorResponse"}]}}, "/logs-service/api/v2a/public/syslog/{id}/deactivate": {"patch": {"tags": ["v2a] Контроллер: Syslog сервера"], "summary": "Деактивация syslog сервера", "description": "PATCH метод REST API: Деактивация syslog сервера", "operationId": "deactivate", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор syslog сервера типа UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Syslog сервер деактивирован"}, "404": {"description": "syslog сервер не найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/logs-service/api/v2a/public/syslog/{id}/activate": {"patch": {"tags": ["v2a] Контроллер: Syslog сервера"], "summary": "Активация syslog сервера", "description": "PATCH метод REST API: Активация syslog сервера", "operationId": "activate", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор syslog сервера типа UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Syslog сервер активирован"}, "404": {"description": "syslog сервер не найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/logs-service/api/v2/public/syslog/{id}/deactivate": {"patch": {"tags": ["v2] Контроллер: Syslog сервера"], "summary": "Деактивация syslog сервера", "description": "PATCH метод REST API: Деактивация syslog сервера", "operationId": "deactivate_1", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор syslog сервера типа UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Syslog сервер деактивирован"}, "404": {"description": "syslog сервер не найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/logs-service/api/v2/public/syslog/{id}/activate": {"patch": {"tags": ["v2] Контроллер: Syslog сервера"], "summary": "Активация syslog сервера", "description": "PATCH метод REST API: Активация syslog сервера", "operationId": "activate_1", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор syslog сервера типа UUID", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Syslog сервер активирован"}, "404": {"description": "syslog сервер не найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/templates-service/api/v2a/public/templates": {"get": {"tags": ["v2a] Контроллер: шаблоны"], "summary": "Получить список шаблонов с фильтрацией, пагинацией и сортировкой", "description": "Get метод REST API получения списка шаблонов с фильтрацией, пагинацией и сортировкой", "operationId": "findAll", "parameters": [{"name": "pageLimit", "in": "query", "description": "Пагинация: ограничение на размер списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "query", "description": "Пагинация: смещение от начала списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "description": "Сортировка: имя поля, по которому выполнять сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortDirection", "in": "query", "description": "Сортировка: направление сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN", "asc", "desc"]}}, {"name": "types", "in": "query", "description": "Фильтр: по типам шаблона", "schema": {"type": "array", "items": {"type": "string", "enum": ["UNKNOWN", "EMBEDDED", "CLONED", "IMPORTED"]}}, {"name": "certificateType", "in": "query", "description": "Фильтр: по типу выпускаемого сертификата", "schema": {"type": "string", "enum": ["UNKNOWN", "EMBEDDED", "CLONED", "IMPORTED"]}}, {"name": "search", "in": "query", "description": "Фильтр: поиск по

```

```
имени", "schema": {"type": "string", "default": "Name"}}, {"name": "removed", "in": "query", "description":
n": "Фильтр: поиск удаленного
шаблона", "schema": {"type": "boolean", "default": false}}, {"name": "id", "in": "query", "description":
"Фильтр: поиск по списку
идентификаторов", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name":
"notId", "in": "query", "description": "Фильтр: Исключая идентификаторы
сертификатов", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "ke
yAlgorithm", "in": "query", "description": "Фильтр: алгоритм ключа
сертификата", "schema": {"type": "array", "items": {"$ref": "#/components/schemas/EKeyAlgorithmV2aPu
blic"}}}], "responses": {"200": {"description": "Список успешно
получен", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseTemplateIt
emV2aResponsePublic"}}}}, "400": {"description": "Ошибка получения
шаблона", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"d
escription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/templ
ates-service/api/v2a/public/templates/{id}": {"get": {"tags": ["v2a] Контроллер:
шаблоны"], "summary": "Получить шаблон по идентификатору", "description": "Get метод REST API
получения шаблона по
идентификатору", "operationId": "findById", "parameters": [{"name": "id", "in": "path", "description":
"Идентификатор
шаблона", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"des
cription": "Шаблон успешно
получен", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseTemplateV2aRespo
nsePublic"}}}}, "400": {"description": "Ошибка получения
шаблона", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"d
escription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/templ
ates-service/api/v2/public/templates": {"get": {"tags": ["v2] Контроллер:
шаблоны"], "summary": "Получить список шаблонов с фильтрацией, пагинацией и
сортировкой", "description": "Get метод REST API получения списка шаблонов с фильтрацией,
пагинацией и
сортировкой", "operationId": "findAll_1", "parameters": [{"name": "pageLimit", "in": "query", "descrip
tion": "Пагинация: ограничение на размер
списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "qu
ery", "description": "Пагинация: смещение от начала
списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "de
scription": "Сортировка: имя поля, по которому выполнять
сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortD
irection", "in": "query", "description": "Сортировка: направление
сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN»,
"asc", "desc"]}}, {"name": "types", "in": "query", "description": "Фильтр: по типам
шаблона", "schema": {"type": "array", "items": {"type": "string", "enum": ["UNKNOWN»,
"EMBEDDED", "CLONED", "IMPORTED"]}}}, {"name": "certificateType", "in": "query", "description": "Фильтр:
по типу выпускаемого сертификата", "schema": {"type": "string", "enum": ["UNKNOWN»,
"EMBEDDED", "CLONED", "IMPORTED"]}}, {"name": "search", "in": "query", "description": "Фильтр: поиск
по
имени", "schema": {"type": "string", "default": "Name"}}, {"name": "removed", "in": "query", "descriptio
n": "Фильтр: поиск удаленного
шаблона", "schema": {"type": "boolean", "default": false}}, {"name": "id", "in": "query", "description":
"Фильтр: поиск по списку
идентификаторов", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name":
"notId", "in": "query", "description": "Фильтр: Исключая идентификаторы
сертификатов", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}], "responses
": {"200": {"description": "Список успешно
получен", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseTemplateIt
emResponsePublic"}}}}, "400": {"description": "Ошибка получения
шаблона", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"d
escription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/templ
ates-service/api/v2/public/templates/{id}": {"get": {"tags": ["v2] Контроллер:
шаблоны"], "summary": "Получить шаблон по идентификатору", "description": "Get метод REST API
```

получения шаблона по идентификатору", "operationId": "findById_1", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор шаблона", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Шаблон успешно получен", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ItemResponseTemplateResponsePublic"}}}}, "400": {"description": "Ошибка получения шаблона", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ErrorResponse"}}}}}, "/subjects-service/api/v2a/public/subjects/{id}": {"get": {"tags": ["v2a Контроллер: субъекты"], "summary": "Поиск субъекта по ID", "description": "GET метод REST API: поиск субъекта по ID", "operationId": "findById_2", "parameters": [{"name": "id", "in": "path", "description": "ID субъекта", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Найденный субъект", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ItemResponseSubjectV2aResponsePublic"}}}}, "404": {"description": "Субъект с указанным ID не существует", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ErrorResponse"}}}}}, "delete": {"tags": ["v2a Контроллер: субъекты"], "summary": "Удаление субъекта по ID", "description": "DELETE метод REST API: удаление субъекта по ID", "operationId": "deleteById", "parameters": [{"name": "id", "in": "path", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Субъект удален"}, "404": {"description": "Субъект с указанным ID не существует", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ErrorResponse"}}}}}, "/subjects-service/api/v2a/public/subjects/ids": {"get": {"tags": ["v2a Контроллер: субъекты"], "summary": "Поиск ID субъектов", "description": "GET метод REST API: поиск ID субъектов", "operationId": "findAllAsIds", "parameters": [{"name": "id", "in": "query", "description": "Фильтр: ID субъекта", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "notId", "in": "query", "description": "Фильтр: исключая ID субъекта", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "resourceId", "in": "query", "description": "Фильтр: ID ресурсной системы", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "securityGroupId", "in": "query", "description": "Фильтр: ID группы безопасности", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "organizationalUnitId", "in": "query", "description": "Фильтр: ID подразделения", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "search", "in": "query", "description": "Фильтр: полнотекстовый поиск по имени", "schema": {"type": "string", "default": "Name"}}, {"name": "isConnected", "in": "query", "description": "Фильтр: подключение субъекта к ресурсной системе", "schema": {"type": "boolean"}}, {"name": "isBlocked", "in": "query", "description": "Фильтр: блокировка субъекта в ресурсной системе", "schema": {"type": "boolean"}}, {"name": "pageLimit", "in": "query", "description": "Пагинация: ограничение на размер списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "query", "description": "Пагинация: смещение от начала списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "description": "Сортировка: имя поля, по которому выполнять сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortDirection", "in": "query", "description": "Сортировка: направление сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN", "asc", "desc"]}}, "responses": {"200": {"description": "Список найденных ID субъектов", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/CollectionResponseUUID"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"\$ref": "#/components/schemas/ErrorResponse"}}}}}, "/subjects-service/api/v2a/public/security-groups": {"get": {"tags": ["v2a Контроллер: группы безопасности"], "summary": "Поиск групп безопасности", "description": "Get метод REST API: поиск

групп

```

безопасности", "operationId": "findAll_3", "parameters": [{"name": "pageLimit", "in": "query", "description": "Пагинация: ограничение на размер списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "query", "description": "Пагинация: смещение от начала списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "description": "Сортировка: имя поля, по которому выполнять сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortDirection", "in": "query", "description": "Сортировка: направление сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN", "asc", "desc"]}}, {"name": "id", "in": "query", "description": "Фильтр: ID группы безопасности", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "notId", "in": "query", "description": "Фильтр: ID группы безопасности (исключающий)", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "resourceId", "in": "query", "description": "Фильтр: ID ресурсной системы", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "subjectId", "in": "query", "description": "Фильтр: ID субъекта", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "search", "in": "query", "description": "Фильтр: полнотекстовый поиск по имени", "schema": {"type": "string", "default": "O=organization"}}, {"name": "isActive", "in": "query", "description": "Фильтр: активная для назначения прав группа безопасности", "schema": {"type": "boolean"}}], "responses": {"200": {"description": "Список найденных групп безопасности", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseSecurityGroupV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/subjects-service/api/v2a/public/security-groups/{groupId}": {"get": {"tags": ["v2a"], "summary": "Поиск группы безопасности по ID", "description": "Get метод REST API: Поиск группы безопасности по ID", "operationId": "findById_3", "parameters": [{"name": "groupId", "in": "path", "description": "ID группы безопасности", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Найденная группа безопасности", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseSecurityGroupV2aResponsePublic"}}}}, "404": {"description": "Группа безопасности с указанным ID не существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/subjects-service/api/v2a/public/resources": {"get": {"tags": ["v2a"], "summary": "Поиск ресурсных систем", "description": "GET метод REST API: поиск ресурсных систем", "operationId": "findAll_4", "parameters": [{"name": "id", "in": "query", "description": "Фильтр: ID ресурсной системы", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "securityGroupId", "in": "query", "description": "Фильтр: ID группы безопасности", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "subjectId", "in": "query", "description": "Фильтр: ID субъекта", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "organizationalUnitId", "in": "query", "description": "Фильтр: ID подразделения", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "search", "in": "query", "description": "Фильтр: полнотекстовый поиск по имени", "schema": {"type": "string", "default": "Name"}}, {"name": "isConnected", "in": "query", "description": "Фильтр: подключенная ресурсная система", "schema": {"type": "boolean"}}, {"name": "isDefault", "in": "query", "description": "Фильтр: ресурсная система по умолчанию", "schema": {"type": "boolean"}}], "responses": {"200": {"description": "Список найденных ресурсных систем", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/CollectionResponseResourceV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/subjects-service/api/v2a/public/resources/{id}": {"get": {"tags": ["v2a"], "summary": "Поиск ресурсной системы", "description": "Get метод REST API: поиск ресурсной системы", "operationId": "getResourceV2a", "parameters": [{"name": "id", "in": "path", "description": "ID ресурсной системы", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Найденная ресурсная система", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ItemResponseResourceV2aResponsePublic"}}}}, "404": {"description": "Ресурсная система с указанным ID не существует", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}
```

```

ресурсные системы"], "summary": "Поиск ресурсной системы по ID", "description": "GET метод REST API: поиск ресурсной системы по ID", "operationId": "findById_4", "parameters": [{"name": "id", "in": "path", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Найденная ресурсная система", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ItemResponseResourceV2aResponsePublic"}}}}, "404": {"description": "Ресурсной системы с указанным ID не существует", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/subjects-service/api/v2a/public/organizational-units": {"get": {"tags": ["v2a] Контроллер: подразделения"], "summary": "Поиск подразделений", "description": "Get метод REST API: поиск подразделений", "operationId": "findAll_5", "parameters": [{"name": "pageLimit", "in": "query", "description": "Пагинация: ограничение на размер списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "query", "description": "Пагинация: смещение от начала списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "description": "Сортировка: имя поля, по которому выполнять сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortDirection", "in": "query", "description": "Сортировка: направление сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN", "asc", "desc"]}}, {"name": "id", "in": "query", "description": "Фильтр: ID подразделения", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "notId", "in": "query", "description": "Фильтр: ID подразделения (исключающий)", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "resourceId", "in": "query", "description": "Фильтр: ID ресурсной системы", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "subjectId", "in": "query", "description": "Фильтр: ID субъекта", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "search", "in": "query", "description": "Фильтр: полнотекстовый поиск по имени", "schema": {"type": "string", "default": "O=organization"}}], "responses": {"200": {"description": "Список найденных подразделений", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseOrganizationalUnitV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/subjects-service/api/v2a/public/organizational-units/{unitId}": {"get": {"tags": ["v2a] Контроллер: подразделения"], "summary": "Поиск подразделения по ID", "description": "Get метод REST API: поиск подразделения по ID", "operationId": "findById_5", "parameters": [{"name": "unitId", "in": "path", "description": "ID подразделения", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Найденное подразделение", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseOrganizationalUnitV2aResponsePublic"}}}}, "404": {"description": "Подразделение с указанным ID не существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/subjects-service/api/v2/public/subjects/{id}": {"get": {"tags": ["v2] Контроллер: субъекты"], "summary": "Поиск субъекта по ID", "description": "GET метод REST API: поиск субъекта по ID", "operationId": "findById_6", "parameters": [{"name": "id", "in": "path", "description": "ID субъекта", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Найденный субъект", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseSubjectResponsePublic"}}}}, "404": {"description": "Субъект с указанным ID не существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"delete": {"tags": ["v2] Контроллер: субъекты"], "summary": "Удаление субъекта по ID", "description": "DELETE метод REST API: удаление субъекта по ID", "operationId": "deleteById_1", "parameters": [{"name": "id", "in": "path", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"204": {"description": "Субъект удален"}, "404": {"description": "Субъект с указанным ID не

```

```

существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500":
{"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/subjects-service/api/v2/public/resources": {"get": {"tags": ["v2] Контроллер: ресурсные
системы"], "summary": "Поиск ресурсных системы", "description": "GET метод REST API: поиск
ресурсных
системы", "operationId": "findAll_7", "parameters": [{"name": "id", "in": "query", "description": "Филь
тр: ID ресурсной
системы", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "securityGroupId", "in": "query", "description": "Фильтр: ID группы
безопасности", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "subjectId", "in": "query", "description": "Фильтр: ID
субъекта", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "organizationalUnitId", "in": "query", "description": "Фильтр: ID
подразделения", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "search", "in": "query", "description": "Фильтр: полнотекстовый поиск по
имени", "schema": {"type": "string", "default": "Name"}}, {"name": "isConnected", "in": "query", "description": "Фильтр: подключенная ресурсная
система", "schema": {"type": "boolean"}}, {"name": "isDefault", "in": "query", "description": "Фильтр:
ресурсная система по
умолчанию", "schema": {"type": "boolean"}}}], "responses": {"200": {"description": "Список найденных
ресурсных
системы", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/CollectionResponseResourceResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/subjects-service/api/v2/public/resources/{id}": {"get": {"tags": ["v2] Контроллер:
ресурсные системы"], "summary": "Поиск ресурсной системы по ID", "description": "GET метод REST
API: поиск ресурсной системы по
ID", "operationId": "findById_7", "parameters": [{"name": "id", "in": "path", "required": true, "schema": {"type": "string", "format": "uuid"}}, {"name": "isDefault", "in": "query", "description": "Фильтр:
ресурсная система по
умолчанию", "schema": {"type": "boolean"}}, {"name": "isConnected", "in": "query", "description": "Фильтр: подключенная ресурсная
система", "schema": {"type": "boolean"}}, {"name": "isDefault", "in": "query", "description": "Фильтр: ресурсная система по
умолчанию", "schema": {"type": "boolean"}}}], "responses": {"200": {"description": "Найденная ресурсная
система", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ItemResponseResourceResponsePublic"}}}}, "404": {"description": "Ресурсной системы с указанным ID не
существует", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/settings-service/api/v2a/public/issuer": {"get": {"tags": ["v2a] Контроллер:
Издатели"], "summary": "Поиск издателей", "description": "GET метод REST API: Поиск
издателей", "operationId": "findAll_8", "parameters": [{"name": "pageLimit", "in": "query", "description": "Пагинация: ограничение на размер
списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "query", "description": "Пагинация: смещение от начала
списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "description": "Сортировка: имя поля, по которому выполнять
сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortDirection", "in": "query", "description": "Сортировка: направление
сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN", "asc", "desc"]}}, {"name": "authorityKeyIdentifier", "in": "query", "description": "Фильтр:
Идентификатор ключа
субъекта", "schema": {"type": "string"}}, {"name": "subjectKeyIdentifier", "in": "query", "description": "Фильтр: Идентификатор ключа
УЦ", "schema": {"type": "string"}}, {"name": "isActive", "in": "query", "description": "Фильтр: Признак
активности", "schema": {"type": "boolean"}}, {"name": "isDefault", "in": "query", "description": "Фильтр: ресурсная система по
умолчанию", "schema": {"type": "boolean"}}, {"name": "isConnected", "in": "query", "description": "Фильтр: подключенная ресурсная
система", "schema": {"type": "boolean"}}, {"name": "isDefault", "in": "query", "description": "Фильтр: ресурсная система по
умолчанию", "schema": {"type": "boolean"}}}], "responses": {"200": {"description": "Издатели", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseIssuerV2aResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/security-service/api/v2a/public/accounts": {"get": {"tags": ["v2a] Контроллер: Учетные
записи"], "summary": "Поиск учетных записей", "description": "GET метод REST API: поиск учетных
записей", "operationId": "findAll_9", "parameters": [{"name": "id", "in": "query", "description": "Фильтр: ID
аккаунта", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "notId", "in": "query", "description": "Фильтр: ID аккаунта

```

```
(исключающий)", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "logins", "in": "query", "description": "Фильтр: поиск по имени", "schema": {"type": "array", "items": {"type": "string"}}}, {"name": "search", "in": "query", "description": "Фильтр: полнотекстовый поиск", "schema": {"type": "string"}}, {"responses": {"200": {"description": "Учетные записи успешно получены", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseAccountV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"security-service/api/v2a/public/accounts/{id}": {"get": {"tags": ["v2a"], "summary": "Получение учетной записи по ID", "description": "GET метод REST API: получения учетной записи по ID", "operationId": "findById_8", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор учетной записи", "required": true, "schema": {"type": "string", "format": "uuid"}}}, {"responses": {"200": {"description": "Учетная запись успешно получена", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseAccountV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"security-service/api/v2a/public/accounts/fingerprint/{fingerprint}": {"get": {"tags": ["v2a"], "summary": "Получение учетной записи по отпечатку", "description": "GET метод REST API: получение учетной записи по отпечатку", "operationId": "getByFingerprint", "parameters": [{"name": "fingerprint", "in": "path", "description": "Отпечаток сертификата", "required": true, "schema": {"type": "string"}}}, {"responses": {"200": {"description": "Данные сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseAccountV2aResponsePublic"}}}}, "403": {"description": "Невозможно выполнить авторизацию с использованием сертификата. Сертификат не привязан к пользователю", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "404": {"description": "Учетная запись не найдена", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"ldap-service/api/v2a/public/connection-points": {"get": {"tags": ["v2a"], "summary": "Поиск точек подключения", "description": "Get метод REST API: поиск точек подключения", "operationId": "findAll_12", "parameters": [{"name": "pageLimit", "in": "query", "description": "Пагинация: ограничение на размер списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "query", "description": "Пагинация: смещение от начала списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "description": "Сортировка: имя поля, по которому выполнять сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortDirection", "in": "query", "description": "Сортировка: направление сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN", "asc", "desc"]}}, {"name": "id", "in": "query", "description": "Фильтр: ID точки подключения", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "resourceId", "in": "query", "description": "Фильтр: ID ресурсной системы", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "search", "in": "query", "description": "Фильтр: полнотекстовый поиск по отображаемому имени", "schema": {"type": "string", "default": "Name"}}}, {"responses": {"200": {"description": "Список найденных точек подключения", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseFindConnectionPointV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, {"ldap-service/api/v2a/public/connection-points/{id}": {"get": {"tags": ["v2a"], "summary": "Получение точки подключения по ID", "description": "Get метод REST API: получение точки подключения по ID", "operationId": "findById_11", "parameters": [{"name": "id", "in": "path", "description": "ID точки подключения", "required": true, "schema": {"type": "string", "format": "uuid"}}}, {"responses": {"200": {"description": "Найденная точка
```

```

подключения", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseFindConnecti
onPointV2aResponsePublic"}}}}, "404": {"description": "Точка подключения с указанным ID не
существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500":
{"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/ldap-
service/api/v2/public/connection-points": {"get": {"tags": ["[v2] Контроллер: Точки подключения
ресурсных систем"], "summary": "Поиск точек подключения", "description": "Get метод REST API:
поиск точек
подключения", "operationId": "findAll_13", "parameters": [{"name": "pageLimit", "in": "query", "descri
ption": "Пагинация: ограничение на размер
списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "qu
ery", "description": "Пагинация: смещение от начала
списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "de
scription": "Сортировка: имя поля, по которому выполнять
сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortD
irection", "in": "query", "description": "Сортировка: направление
сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN»,
"asc", "desc"]}}, {"name": "id", "in": "query", "description": "Фильтр: ID точки
подключения", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "res
ourceId", "in": "query", "description": "Фильтр: ID ресурсной
системы", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "search
", "in": "query", "description": "Фильтр: полнотекстовый поиск по отображаемому
имени", "schema": {"type": "string", "default": "Name"}}}, "responses": {"200": {"description": "Список
найденных точек
подключения", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseFindCo
nnectionPointResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/ldap-
service/api/v2/public/connection-points/{id}": {"get": {"tags": ["[v2] Контроллер: Точки
подключения ресурсных систем"], "summary": "Получение точки подключения по
ID", "description": "Get метод REST API: получение точки подключения по
ID", "operationId": "findById_12", "parameters": [{"name": "id", "in": "path", "description": "ID точки
подключения", "required": true, "schema": {"type": "string", "format": "uuid"}}}, "responses": {"200": {
"description": "Найденная точка
подключения", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseFindConnecti
onPointResponsePublic"}}}}, "404": {"description": "Точка подключения с указанным ID не
существует", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500":
{"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/exter
nal-integration-service/api/version": {"get": {"tags": ["version-
controller"], "summary": "Получение версии
модуля", "operationId": "getApiVersion", "responses": {"200": {"description": "Версия
модуля", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/GetApiVersionResponse"}}}}}}
, "/export-
service/api/v2a/public/export/certificates/{certificateId}/pkcs12": {"get": {"tags": ["[v2a]
Контроллер экспорта"], "summary": "Получение файла контейнера P12 по ID
сертификата", "description": "GET метод REST API: Получение файла контейнера
P12", "operationId": "findP12ById", "parameters": [{"name": "certificateId", "in": "path", "descriptio
n": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}}, "responses": {"200": {
"description": "Файл контейнера
P12", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"description": "Вну
тренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2a/public/export/certificates/{certificateId}/pkcs10": {"get": {"tags": ["[v2a]
Контроллер экспорта"], "summary": "Получение файла запроса на сертификата по ID
сертификата", "description": "GET метод REST API: Получение файла запроса на
сертификата", "operationId": "findRequestById", "parameters": [{"name": "certificateId", "in": "path"
, "description": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}}, "responses": {"200": {
"description": "Файл запроса на
сертификат", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"descriptio

```

```

n": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/export-
service/api/v2a/public/export/certificates/{certificateId}/chain": {"get": {"tags": ["v2a]
Контроллер экспорта"], "summary": "Получение файла цепочки сертификатов по ID
сертификата", "description": "GET метод REST API: Получение файла цепочки
сертификатов", "operationId": "findChainById", "parameters": [{"name": "certificateId", "in": "path",
"description": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {
"description": "Файл цепочки
сертификатов", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"descript
ion": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-
service/api/v2a/public/export/certificates/{certificateId}/certificate": {"get": {"tags": ["v2a]
Контроллер экспорта"], "summary": "Получение файла сертификата по ID
сертификата", "description": "GET метод REST API: Получение файла
сертификата", "operationId": "findCertificateById", "parameters": [{"name": "certificateId", "in": "p
ath", "description": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {
"description": "Файл
сертификата", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"descripti
on": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2a/public/export/certificate-authorities/{caId}/delta-
crl": {"get": {"tags": ["v2a] Контроллер экспорта"], "summary": "Получение файла Delta CRL по ID
ЦС", "description": "GET метод REST API: Получение файла Delta
CRL", "operationId": "findDeltaCrlByCaId", "parameters": [{"name": "caId", "in": "path", "description"
: "Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"descript
ion": "Файл Delta
CRL", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"description": "Вну
тренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2a/public/export/certificate-authorities/{caId}/crl": {"get": {"tags": ["v2a]
Контроллер экспорта"], "summary": "Получение файла CRL по ID ЦС", "description": "GET метод REST
API: Получение файла
CRL", "operationId": "findCrlByCaId", "parameters": [{"name": "caId", "in": "path", "description": "Иде
нтификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"descript
ion": "Файл
CRL", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"description": "Вну
тренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2a/public/export/certificate-authorities/{caId}/chain": {"get": {"tags": ["v2a]
Контроллер экспорта"], "summary": "Получение файла цепочки сертификатов по ID
ЦС", "description": "GET метод REST API: Получение файла цепочки
сертификатов", "operationId": "findChainByCaId", "parameters": [{"name": "caId", "in": "path", "descri
ption": "Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"descript
ion": "Файл цепочки
сертификатов", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"descript
ion": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2a/public/export/certificate-
authorities/{caId}/certificate": {"get": {"tags": ["v2a] Контроллер
экспорта"], "summary": "Получение файла сертификата по ID ЦС", "description": "GET метод REST API:
Получение файла
сертификата", "operationId": "findCertificateByCaId", "parameters": [{"name": "caId", "in": "path", "d
escription": "Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"descript
ion": "Файл

```

```

сертификата", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/export-
service/api/v2/public/export/certificates/{certificateId}/pkcs12": {"get": {"tags": ["v2]
Контроллер экспорта"], "summary": "Получение файла контейнера P12 по ID
сертификата", "description": "GET метод REST API: Получение файла контейнера
P12", "operationId": "findP12ById_1", "parameters": [{"name": "certificateId", "in": "path", "description": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"
description": "Файл контейнера
P12", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"description": "Вну
тренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2/public/export/certificates/{certificateId}/pkcs10": {"get": {"tags": ["v2]
Контроллер экспорта"], "summary": "Получение файла запроса на сертификата по ID
сертификата", "description": "GET метод REST API: Получение файла запроса на
сертификата", "operationId": "findRequestById_1", "parameters": [{"name": "certificateId", "in": "pat
h", "description": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"
description": "Файл запроса на
сертификат", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"descriptio
n": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2/public/export/certificates/{certificateId}/chain": {"get": {"tags": ["v2]
Контроллер экспорта"], "summary": "Получение файла цепочки сертификатов по ID
сертификата", "description": "GET метод REST API: Получение файла цепочки
сертификатов", "operationId": "findChainById_1", "parameters": [{"name": "certificateId", "in": "path
", "description": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"
description": "Файл цепочки
сертификатов", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"descripti
on": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2/public/export/certificates/{certificateId}/certificate": {"get": {"tags": ["v2]
Контроллер экспорта"], "summary": "Получение файла сертификата по ID
сертификата", "description": "GET метод REST API: Получение файла
сертификата", "operationId": "findCertificateById_1", "parameters": [{"name": "certificateId", "in":
"path", "description": "Идентификатор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"
description": "Файл
сертификата", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"descripti
on": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2/public/export/certificate-authorities/{caId}/delta-crl": {"get": {"tags": ["v2]
Контроллер экспорта"], "summary": "Получение файла Delta CRL по ID ЦС", "description": "GET метод
REST API: Получение файла Delta
CRL", "operationId": "findDeltaCrlByCaId_1", "parameters": [{"name": "caId", "in": "path", "descriptio
n": "Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"descript
ion": "Файл Delta
CRL", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"description": "Вну
тренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2/public/export/certificate-authorities/{caId}/crl": {"get": {"tags": ["v2]
Контроллер экспорта"], "summary": "Получение файла CRL по ID ЦС", "description": "GET метод REST
API: Получение файла
CRL", "operationId": "findCrlByCaId_1", "parameters": [{"name": "caId", "in": "path", "description": "И
дентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"descript
ion": "Файл
CRL", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}}, "500": {"description": "Вну

```

тренингов

```

ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/export-
service/api/v2/public/export/certificate-authorities/{caId}/chain": {"get": {"tags": ["v2]
Контроллер экспорта"], "summary": "Получение файла цепочки сертификатов по ID
ЦС", "description": "GET метод REST API: Получение файла цепочки
сертификатов", "operationId": "findChainByCaId_1", "parameters": [{"name": "caId", "in": "path", "desc
ription": "Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"descript
ion": "Файл цепочки
сертификатов", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}, "500": {"descript
ion": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/expor
t-service/api/v2/public/export/certificate-
authorities/{caId}/certificate": {"get": {"tags": ["v2] Контроллер
экспорта"], "summary": "Получение файла сертификата по ID ЦС", "description": "GET метод REST API:
Получение файла
сертификата", "operationId": "findCertificateByCaId_1", "parameters": [{"name": "caId", "in": "path",
"description": "Идентификатор
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"descript
ion": "Файл
сертификата", "content": {"*//*": {"schema": {"type": "string", "format": "byte"}}}, "500": {"descripti
on": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2a/public/certificates": {"get": {"tags": ["v2a] Контроллер:
сертификаты"], "summary": "Получение списка сертификатов", "description": "GET метод REST API:
получение списка
сертификатов", "operationId": "findAll_14", "parameters": [{"name": "pageLimit", "in": "query", "descr
iption": "Пагинация: ограничение на размер
списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "qu
ery", "description": "Пагинация: смещение от начала
списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "de
scription": "Сортировка: имя поля, по которому выполнять
сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortD
irection", "in": "query", "description": "Сортировка: направление
сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN",
"asc", "desc"]}}, {"name": "search", "in": "query", "description": "Фильтр: часть имени или серийного
номера
сертификата", "schema": {"type": "string"}}, {"name": "issuerId", "in": "query", "description": "Фильтр
: уникальный идентификатор издающего
сертификата", "schema": {"type": "string", "format": "uuid"}}, {"name": "templateId", "in": "query", "de
scription": "Фильтр: идентификатор шаблона, по которому выпущен
сертификат", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "stat
us", "in": "query", "description": "Фильтр: статус
сертификата", "schema": {"type": "array", "items": {"type": "string", "description": "Перечисление:
статус сертификата", "enum": ["UNKNOWN",
"ACTIVE", "HOLD", "REVOKE", "REQUEST"]}}}, {"name": "type", "in": "query", "description": "Фильтр: тип
сертификата", "schema": {"type": "array", "items": {"type": "string", "description": "Перечисление
типов сертификата", "enum": ["UNKNOWN",
"CERTIFICATE", "ROOT_CA", "SUB_CA"]}}}, {"name": "revocationReason", "in": "query", "description": "Фи
льтр: причина отзыва
(соответствие)", "schema": {"type": "array", "items": {"type": "string", "description": "Коды словаря:
причины отзыва", "enum": ["UNKNOWN",
"UNSPECIFIED", "KEY_COMPROMISE", "CA_COMPROMISE", "AFFILIATION_CHANGED", "SUPERSEDED", "CESSATION_O
F_OPERATION", "CERTIFICATE_HOLD", "REMOVE_FROM_CRL", "PRIVILEGE_WITHDRAWN", "AA_COMPROMISE"]}}}, {"
name": "notRevocationReason", "in": "query", "description": "Фильтр: причина отзыва
(отсутствие)", "schema": {"type": "array", "items": {"type": "string", "description": "Коды словаря:
причины отзыва", "enum": ["UNKNOWN",
"UNSPECIFIED", "KEY_COMPROMISE", "CA_COMPROMISE", "AFFILIATION_CHANGED", "SUPERSEDED", "CESSATION_O
F_OPERATION", "CERTIFICATE_HOLD", "REMOVE_FROM_CRL", "PRIVILEGE_WITHDRAWN", "AA_COMPROMISE"]}}}, {"
name": "revocationDateFrom", "in": "query", "description": "Фильтр: дата отзыва
(начало)", "schema": {"type": "string", "format": "date-

```

```

time"}}, {"name": "revocationDateTo", "in": "query", "description": "Фильтр: дата отзыва (конец)", "schema": {"type": "string", "format": "date-time"}}, {"name": "hasRevocationReason", "in": "query", "description": "Фильтр: причина отзыва (наличие)", "schema": {"type": "boolean"}}, {"name": "hasRequest", "in": "query", "description": "Фильтр: запрос сертификата (наличие)", "schema": {"type": "boolean"}}, {"name": "hasCA", "in": "query", "description": "Фильтр: ЦС (наличие)", "schema": {"type": "boolean"}}, {"name": "isManagementCA", "in": "query", "description": "Фильтр: ЦС (технологический)", "schema": {"type": "boolean"}}, {"name": "isValid", "in": "query", "description": "Фильтр: сертификат действителен", "schema": {"type": "boolean"}}, {"name": "isExpired", "in": "query", "description": "Фильтр: сертификат истек", "schema": {"type": "boolean"}}, {"name": "validFromFrom", "in": "query", "description": "Фильтр: дата начала действия сертификата (начало)", "schema": {"type": "string", "format": "date-time"}}, {"name": "validFromTo", "in": "query", "description": "Фильтр: дата начала действия сертификата (конец)", "schema": {"type": "string", "format": "date-time"}}, {"name": "validToFrom", "in": "query", "description": "Фильтр: дата окончания действия сертификата (начало)", "schema": {"type": "string", "format": "date-time"}}, {"name": "validToTo", "in": "query", "description": "Фильтр: дата окончания действия сертификата (конец)", "schema": {"type": "string", "format": "date-time"}}, {"name": "updatedFrom", "in": "query", "description": "Фильтр: дата обновления сертификата (начало)", "schema": {"type": "string", "format": "date-time"}}, {"name": "updatedTo", "in": "query", "description": "Фильтр: дата обновления сертификата (конец)", "schema": {"type": "string", "format": "date-time"}}, {"name": "subjectId", "in": "query", "description": "Фильтр: Идентификаторы субъектов", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "userId", "in": "query", "description": "Фильтр: Идентификаторы пользователей", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "serialnumber", "in": "query", "description": "Фильтр: Серийные номера сертификатов", "schema": {"type": "array", "items": {"type": "string"}}}, {"name": "fingerprint", "in": "query", "description": "Фильтр: Отпечаток сертификата", "schema": {"type": "array", "items": {"type": "string"}}}, {"name": "subjectKeyIdentifier", "in": "query", "description": "Фильтр: Идентификатор ключа субъекта", "schema": {"type": "array", "items": {"type": "string"}}}, {"name": "notId", "in": "query", "description": "Фильтр: Исключая идентификаторы сертификатов", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}], "responses": {"200": {"description": "Список сертификатов", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseFindAllCertificateV2aResponsePublic"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}, "/certificate-authority-service/api/v2a/public/certificates/{id}": {"get": {"tags": ["v2a"], "summary": "Получение сертификата по id", "description": "GET метод REST API: получение сертификата по id", "operationId": "findById_13", "parameters": [{"name": "id", "in": "path", "description": "Идентификатор сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}], "responses": {"200": {"description": "Данные сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseCertificateV2aResponsePublic"}}}}, "404": {"description": "Сертификат не найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"description": "Внутренняя ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certificate-authority-service/api/v2a/public/certificates/serialNumber/{serialNumber}": {"get": {"tags": ["v2a"], "summary": "Получение сертификата по серийному номеру", "description": "GET метод REST API: получение сертификата по серийному номеру", "operationId": "getBySerialNumber", "parameters": [{"name": "serialNumber", "in": "path", "description": "Серийный номер сертификата", "required": true, "schema": {"type": "string"}}], "responses": {"200": {"description": "Данные сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseCertificateV

```

```

2aResponsePublic"}}}}}, "404": {"description": "Сертификат не
найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-
service/api/v2a/public/certificates/fingerprint/{fingerprint}": {"get": {"tags": ["v2a]
Контроллер: сертификаты", "summary": "Получение сертификата по отпечатку", "description": "GET
метод REST API: получение сертификата по
отпечатку", "operationId": "getByFingerprint_1", "parameters": [{"name": "fingerprint", "in": "path",
"description": "Отпечаток
сертификата", "required": true, "schema": {"type": "string"}}], "responses": {"200": {"description": "Д
анные
сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseCertificateV
2aResponsePublic"}}}}, "404": {"description": "Сертификат не
найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2a/public/certificate-authorities": {"get": {"tags": ["v2a]
Контроллер: Центры сертификации", "summary": "Получение списка ЦС", "description": "GET метод
REST API: Получение списка
ЦС", "operationId": "findAll_15", "parameters": [{"name": "pageLimit", "in": "query", "description": "П
агинация: ограничение на размер
списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "qu
ery", "description": "Пагинация: смещение от начала
списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "de
scription": "Сортировка: имя поля, по которому выполнять
сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortD
irection", "in": "query", "description": "Сортировка: направление
сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN",
"asc", "desc"]}}, {"name": "search", "in": "query", "description": "Фильтр: Отличительное имя
ЦС", "schema": {"type": "string"}}, {"name": "status", "in": "query", "description": "Фильтр: статус
сертификата ЦС", "schema": {"type": "array", "items": {"type": "string", "description": "Перечисление:
статус сертификата", "enum": ["UNKNOWN",
"ACTIVE", "HOLD", "REVOKE", "REQUEST"]}}}, {"name": "type", "in": "query", "description": "Фильтр: тип
сертификата ЦС", "schema": {"type": "array", "items": {"type": "string", "description": "Перечисление
типов сертификата", "enum": ["UNKNOWN",
"CERTIFICATE", "ROOT_CA", "SUB_CA"]}}}, {"name": "isManagement", "in": "query", "description": "Фильтр
: ЦС
(технологический)", "schema": {"type": "boolean"}}, {"name": "isActive", "in": "query", "description":
"Фильтр: ЦС
(технологический)", "schema": {"type": "boolean"}}, {"name": "isValid", "in": "query", "description": "
Фильтр: сертификат ЦС
действителен", "schema": {"type": "boolean"}}, {"name": "isExpired", "in": "query", "description": "Фил
ьтр: сертификат ЦС
истек", "schema": {"type": "boolean"}}, {"name": "id", "in": "query", "description": "Фильтр: по
идентификаторам
ЦС", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "notIds", "in"
: "query", "description": "Фильтр: Исключая идентификаторы
ЦС", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}], "responses": {"200": {
"description": "Список
ЦС", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseFindCertificate
AuthorityV2aResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2a/public/certificate-authorities/{caId}": {"get": {"tags": ["v2a]
Контроллер: Центры сертификации", "summary": "Получение ЦС", "description": "GET метод REST API:
Получение
ЦС", "operationId": "findById_14", "parameters": [{"name": "caId", "in": "path", "description": "ID
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}, {"name": "id", "in": "query", "de
scription": "Фильтр: по идентификаторам
ЦС", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}], "responses": {"200": {"descript
ion": "ЦС", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseFindCertificate
AuthorityV2aResponsePublic"}}}}, "404": {"description": "ЦС не
найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de

```

```

scription":"Внутренняя
ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}},"/certi
ficate-authority-service/api/v2a/public/certificate-authorities/active":{"get":{"tags":["[v2a]
Контроллер: Центры сертификации"],"summary":"Получение активного ЦС","description":"GET метод
REST API: Получение активного
ЦС","operationId":"active","responses":{"200":{"description":"Активный
ЦС","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ItemResponseFindCertificateAuthor
ityV2aResponsePublic"}}}},"404":{"description":"Активный ЦС не
найден","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}},"500":{"de
scription":"Внутренняя
ошибка","content":{"*//*":{"schema":{"$ref":"#/components/schemas/ErrorResponse"}}}}}}},"/certi
ficate-authority-service/api/v2/public/certificates":{"get":{"tags":["[v2] Контроллер:
сертификаты"],"summary":"Получение списка сертификатов","description":"GET метод REST API:
получение списка
сертификатов","operationId":"findAll_16","parameters":[{"name":"pageLimit","in":"query","descr
iption":"Пагинация: ограничение на размер
списка","schema":{"minimum":0,"type":"integer","default":10000}},{"name":"pageOffset","in":"qu
ery","description":"Пагинация: смещение от начала
списка","schema":{"minimum":0,"type":"integer","default":0}},{"name":"sortBy","in":"query","de
scription":"Сортировка: имя поля, по которому выполнять
сортировку","schema":{"type":"array","items":{"type":"string","default":"id"}},{"name":"sortD
irection","in":"query","description":"Сортировка: направление
сортировки","schema":{"type":"string","default":"asc","enum":["UNKNOWN",
"asc","desc"]}],{"name":"search","in":"query","description":"Фильтр: часть имени или серийного
номера
сертификата","schema":{"type":"string"}}, {"name":"issuerId","in":"query","description":"Фильтр
: уникальный идентификатор издающего
сертификата","schema":{"type":"string","format":"uuid"}}, {"name":"templateId","in":"query","de
scription":"Фильтр: идентификатор шаблона, по которому выпущен
сертификат","schema":{"type":"array","items":{"type":"string","format":"uuid"}}, {"name":"stat
us","in":"query","description":"Фильтр: статус
сертификата","schema":{"type":"array","items":{"type":"string","description":"Перечисление:
статус сертификата","enum":["UNKNOWN»,
"ACTIVE","HOLD","REVOKE","REQUEST"]}}}, {"name":"type","in":"query","description":"Фильтр: тип
сертификата","schema":{"type":"array","items":{"type":"string","description":"Перечисление
типов сертификата","enum":["UNKNOWN»,
"CERTIFICATE","ROOT_CA","SUB_CA"]}}}, {"name":"revocationReason","in":"query","description":"Фи
льтр: причина отзыва
(соответствие)","schema":{"type":"array","items":{"type":"string","description":"Коды словаря:
причины отзыва","enum":["UNKNOWN»,
"UNSPECIFIED","KEY_COMPROMISE","CA_COMPROMISE","AFFILIATION_CHANGED","SUPERSEDED","CESSATION_O
F_OPERATION","CERTIFICATE_HOLD","REMOVE_FROM_CRL","PRIVILEGE_WITHDRAWN","AA_COMPROMISE"]}}}, {"
name":"notRevocationReason","in":"query","description":"Фильтр: причина отзыва
(отсутствии)","schema":{"type":"array","items":{"type":"string","description":"Коды словаря:
причины отзыва","enum":["UNKNOWN»,
"UNSPECIFIED","KEY_COMPROMISE","CA_COMPROMISE","AFFILIATION_CHANGED","SUPERSEDED","CESSATION_O
F_OPERATION","CERTIFICATE_HOLD","REMOVE_FROM_CRL","PRIVILEGE_WITHDRAWN","AA_COMPROMISE"]}}}, {"
name":"revocationDateFrom","in":"query","description":"Фильтр: дата отзыва
(начало)","schema":{"type":"string","format":"date-
time"}}, {"name":"revocationDateTo","in":"query","description":"Фильтр: дата отзыва
(конец)","schema":{"type":"string","format":"date-
time"}}, {"name":"hasRevocationReason","in":"query","description":"Фильтр: причина отзыва
(наличие)","schema":{"type":"boolean"}}, {"name":"hasRequest","in":"query","description":"Фильт
р: запрос сертификата
(наличие)","schema":{"type":"boolean"}}, {"name":"hasCA","in":"query","description":"Фильтр: ЦС
(наличие)","schema":{"type":"boolean"}}, {"name":"isManagementCA","in":"query","description":"Ф
ильтр: ЦС
(технологический)","schema":{"type":"boolean"}}, {"name":"isValid","in":"query","description":"
Фильтр: сертификат
действителен","schema":{"type":"boolean"}}, {"name":"isExpired","in":"query","description":"Фил
ьтр: сертификат

```

```

истек", "schema": {"type": "boolean"}}, {"name": "validFromFrom", "in": "query", "description": "Фильтр
: дата начала действия сертификата (начало)", "schema": {"type": "string", "format": "date-
time"}}, {"name": "validFromTo", "in": "query", "description": "Фильтр: дата начала действия
сертификата (конец)", "schema": {"type": "string", "format": "date-
time"}}, {"name": "validToFrom", "in": "query", "description": "Фильтр: дата окончания действия
сертификата (начало)", "schema": {"type": "string", "format": "date-
time"}}, {"name": "validToTo", "in": "query", "description": "Фильтр: дата окончания действия
сертификата (конец)", "schema": {"type": "string", "format": "date-
time"}}, {"name": "updatedFrom", "in": "query", "description": "Фильтр: дата обновления сертификата
(начало)", "schema": {"type": "string", "format": "date-
time"}}, {"name": "updatedTo", "in": "query", "description": "Фильтр: дата обновления сертификата
(конец)", "schema": {"type": "string", "format": "date-
time"}}, {"name": "subjectId", "in": "query", "description": "Фильтр: Идентификаторы
субъектов", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "userI
d", "in": "query", "description": "Фильтр: Идентификаторы
пользователей", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}, {"name": "s
erialnumber", "in": "query", "description": "Фильтр: Серийные номера
сертификатов", "schema": {"type": "array", "items": {"type": "string"}}}, {"name": "fingerprint", "in":
"query", "description": "Фильтр: Отпечаток
сертификата", "schema": {"type": "array", "items": {"type": "string"}}}, {"name": "subjectKeyIdentifie
r", "in": "query", "description": "Фильтр: Идентификатор ключа
субъекта", "schema": {"type": "array", "items": {"type": "string"}}}, {"name": "notId", "in": "query", "d
escription": "Фильтр: Исключая идентификаторы
сертификатов", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}], "responses
": {"200": {"description": "Список
сертификатов", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/CollectionResponseFindA
llCertificateResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2/public/certificates/{id}": {"get": {"tags": ["v2] Контроллер:
сертификаты"], "summary": "Получение сертификата по id", "description": "GET метод REST API:
получение сертификата по
id", "operationId": "findById_15", "parameters": [{"name": "id", "in": "path", "description": "Идентифи
катор
сертификата", "required": true, "schema": {"type": "string", "format": "uuid"}}}], "responses": {"200": {
"description": "Данные
сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseCertificateR
esponsePublic"}}}}, "404": {"description": "Сертификат не
найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-
service/api/v2/public/certificates/serialNumber/{serialNumber}": {"get": {"tags": ["v2]
Контроллер: сертификаты"], "summary": "Получение сертификата по серийному
номеру", "description": "GET метод REST API: получение сертификата по серийному
номеру", "operationId": "getBySerialNumber_1", "parameters": [{"name": "serialNumber", "in": "path", "
description": "Серийный номер
сертификата", "required": true, "schema": {"type": "string"}}}], "responses": {"200": {"description": "Д
анные
сертификата", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ItemResponseCertificateR
esponsePublic"}}}}, "404": {"description": "Сертификат не
найден", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-
service/api/v2/public/certificates/fingerprint/{fingerprint}": {"get": {"tags": ["v2]
Контроллер: сертификаты"], "summary": "Получение сертификата по отпечатку", "description": "GET
метод REST API: получение сертификата по
отпечатку", "operationId": "getByFingerprint_2", "parameters": [{"name": "fingerprint", "in": "path",
"description": "Отпечаток
сертификата", "required": true, "schema": {"type": "string"}}}], "responses": {"200": {"description": "Д
анные

```

```

сертификата", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ItemResponseCertificateR
esponsePublic"}}}}, "404": {"description": "Сертификат не
найден", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2/public/certificate-authorities": {"get": {"tags": ["v2]
Контроллер: Центры сертификации"], "summary": "Получение списка ЦС", "description": "GET метод
REST API: Получение списка
ЦС", "operationId": "findAll_17", "parameters": [{"name": "pageLimit", "in": "query", "description": "П
агинация: ограничение на размер
списка", "schema": {"minimum": 0, "type": "integer", "default": 10000}}, {"name": "pageOffset", "in": "qu
ery", "description": "Пагинация: смещение от начала
списка", "schema": {"minimum": 0, "type": "integer", "default": 0}}, {"name": "sortBy", "in": "query", "de
scription": "Сортировка: имя поля, по которому выполнять
сортировку", "schema": {"type": "array", "items": {"type": "string", "default": "id"}}}, {"name": "sortD
irection", "in": "query", "description": "Сортировка: направление
сортировки", "schema": {"type": "string", "default": "asc", "enum": ["UNKNOWN»,
"asc", "desc"]}}, {"name": "search", "in": "query", "description": "Фильтр: Отличительное имя
ЦС", "schema": {"type": "string"}}, {"name": "status", "in": "query", "description": "Фильтр: статус
сертификата ЦС", "schema": {"type": "array", "items": {"type": "string", "description": "Перечисление:
статус сертификата", "enum": ["UNKNOWN»,
"ACTIVE", "HOLD", "REVOKE", "REQUEST"]}}, {"name": "type", "in": "query", "description": "Фильтр: тип
сертификата ЦС", "schema": {"type": "array", "items": {"type": "string", "description": "Перечисление
типов сертификата", "enum": ["UNKNOWN»,
"CERTIFICATE", "ROOT_CA", "SUB_CA"]}}, {"name": "isManagement", "in": "query", "description": "Фильтр
: ЦС
(технологический)", "schema": {"type": "boolean"}}, {"name": "isActive", "in": "query", "description":
"Фильтр: ЦС
(технологический)", "schema": {"type": "boolean"}}, {"name": "isValid", "in": "query", "description": "
Фильтр: сертификат ЦС
действителен", "schema": {"type": "boolean"}}, {"name": "isExpired", "in": "query", "description": "Фил
ьтр: сертификат ЦС
истек", "schema": {"type": "boolean"}}, {"name": "notIds", "in": "query", "description": "Фильтр:
Исключая идентификаторы
ЦС", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}], "responses": {"200": {
"description": "Список
ЦС", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/CollectionResponseFindCertificate
AuthorityResponsePublic"}}}}, "500": {"description": "Внутренняя
ошибка", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2/public/certificate-authorities/{caId}": {"get": {"tags": ["v2]
Контроллер: Центры сертификации"], "summary": "Получение ЦС", "description": "GET метод REST API:
Получение
ЦС", "operationId": "findById_16", "parameters": [{"name": "caId", "in": "path", "description": "ID
ЦС", "required": true, "schema": {"type": "string", "format": "uuid"}}, {"name": "type", "in": "query", "description": "Фильтр: тип
сертификата ЦС", "schema": {"type": "array", "items": {"type": "string", "description": "Перечисление
типов сертификата", "enum": ["UNKNOWN»,
"CERTIFICATE", "ROOT_CA", "SUB_CA"]}}, {"name": "isManagement", "in": "query", "description": "Фильтр
: ЦС
(технологический)", "schema": {"type": "boolean"}}, {"name": "isActive", "in": "query", "description":
"Фильтр: ЦС
(технологический)", "schema": {"type": "boolean"}}, {"name": "isValid", "in": "query", "description": "
Фильтр: сертификат ЦС
действителен", "schema": {"type": "boolean"}}, {"name": "isExpired", "in": "query", "description": "Фил
ьтр: сертификат ЦС
истек", "schema": {"type": "boolean"}}, {"name": "notIds", "in": "query", "description": "Фильтр:
Исключая идентификаторы
ЦС", "schema": {"type": "array", "items": {"type": "string", "format": "uuid"}}}], "responses": {"200": {
"description": "Список
ЦС", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/CollectionResponseFindCertificate
AuthorityResponsePublic"}}}}, "404": {"description": "ЦС не
найден", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/certi
ficate-authority-service/api/v2/public/certificate-authorities/active": {"get": {"tags": ["v2]
Контроллер: Центры сертификации"], "summary": "Получение активного ЦС", "description": "GET метод
REST API: Получение активного
ЦС", "operationId": "active_1", "responses": {"200": {"description": "Активный
ЦС", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ItemResponseFindCertificateAuthor
ityResponsePublic"}}}}, "404": {"description": "Активный ЦС не
найден", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}, "500": {"de
scription": "Внутренняя
ошибка", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/ErrorResponse"}}}}}}, "/backw
ard-compatibility-service/api/version": {"get": {"tags": ["version-
controller"], "summary": "Получение версии
модуля", "operationId": "getApiVersion_1", "responses": {"200": {"description": "Версия

```

```

модуля", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/VersionResponse"}}}}}}}, "com
ponents": {"schemas": {"Data": {"type": "object", "properties": {"id": {"type": "object", "description"
: "Идентификатор сущности, работа с которым привела к
ошибке"}, "fieldName": {"type": "string", "description": "Имя поля, которое не прошло
валидацию"}, "fieldValue": {"type": "object", "description": "Значение поля, которое не прошло
валидацию"}, "causedBy": {"type": "string", "description": "Сообщение, чем вызвано
исключение"}}, "description": "Объект с деталями
ошибки"}, "ErrorResponse": {"type": "object", "properties": {"status": {"type": "integer", "format": "i
nt32"}, "code": {"type": "integer", "description": "Код ошибки xxxууу, где xxx – код сервиса, ууу –
внутренний код ошибки", "format": "int32"}, "message": {"type": "string", "description": "Описание
ошибки"}, "data": {"$ref": "#/components/schemas/Data"}}, "description": "Стандартизированный
объект ответа сервиса, содержащий информацию об
ошибке"}, "EditSubjectV2aRequestPublic": {"type": "object", "properties": {"id": {"type": "string", "d
escription": "ID
субъекта", "format": "uuid"}, "subjectName": {"type": "object", "additionalProperties": {"type": "arra
y", "description": "Отличительное имя
субъекта", "items": {"type": "string", "description": "Отличительное имя
субъекта"}}, "description": "Отличительное имя
субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "descripti
on": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя
субъекта"}}, "description": "Дополнительное имя субъекта"}}, "description": "Запрос: создание
(обновление)
субъекта"}, "ItemResponseSubjectV2aResponsePublic": {"type": "object", "properties": {"status": {"ty
pe": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/SubjectV2aResponsePublic"
}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный
элемент"}, "ResourceItemV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string"
, "description": "ID ресурсной
системы", "format": "uuid"}, "commonName": {"type": "string", "description": "Общее наименование (CN)
ресурсной системы"}, "distinguishedName": {"type": "string", "description": "Отличающиеся имя
ресурсной системы"}}, "description": "Ответ: ресурсная система
(сокращенный)"}, "SubjectNameFieldItemV2aResponsePublic": {"type": "object", "properties": {"values
": {"type": "array", "description": "Значения атрибута
субъекта", "items": {"type": "string", "description": "Значения атрибута
субъекта"}}, "editable": {"type": "boolean", "description": "Флаг: редактируемый
атрибут"}}, "description": "Ответ: данные атрибута
субъекта"}, "SubjectV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "de
scription": "ID субъекта", "format": "uuid"}, "commonName": {"type": "string", "description": "Общее
наименование (CN) субъекта"}, "distinguishedName": {"type": "string", "description": "Отличающееся
наименование (DN)
субъекта"}, "resource": {"$ref": "#/components/schemas/ResourceItemV2aResponsePublic"}, "subjectNa
me": {"type": "object", "additionalProperties": {"$ref": "#/components/schemas/SubjectNameFieldItem
V2aResponsePublic"}, "description": "Отличительное имя
субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"$ref": "#/components/sche
mas/SubjectNameFieldItemV2aResponsePublic"}, "description": "Дополнительное имя
субъекта"}, "isConnected": {"type": "boolean", "description": "Флаг: подключение субъекта к
ресурсной системе"}, "isBlocked": {"type": "boolean", "description": "Флаг: блокировка субъекта в
ресурсной системе"}, "modify": {"type": "string", "description": "Дата и время обновления в
ресурсной системе", "format": "date-
time"}, "certificatesCount": {"type": "integer", "description": "Количество
сертификатов", "format": "int64"}, "updated": {"type": "string", "description": "Дата и время
обновления", "format": "date-time"}, "created": {"type": "string", "description": "Дата и время
создания", "format": "date-time"}}, "description": "Ответ:
субъект"}, "EditSubjectByCsrV2aRequestPublicMultipartFile": {"required": ["request"], "type": "obje
ct", "properties": {"id": {"type": "string", "description": "ID
субъекта", "format": "uuid"}, "request": {"type": "string", "format": "binary"}, "subjectAltName": {"ty
pe": "object", "additionalProperties": {"type": "array", "description": "Дополнительное имя
субъекта", "items": {"type": "string", "description": "Дополнительное имя
субъекта"}}, "description": "Дополнительное имя субъекта"}}, "description": "Запрос: создание
(обновление) субъекта по запросу
#PKCS10"}, "EditSubjectByCsrV2aRequestPublicFileDescription": {"required": ["request"], "type": "ob
ject", "properties": {"id": {"type": "string", "description": "ID

```

```

субъекта", "format": "uuid", "request": {"$ref": "#/components/schemas/FileDescription"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя субъекта"}}, "description": "Дополнительное имя субъекта"}}, "description": "Запрос: создание (обновление) субъекта по запросу #PKCS10"}, "FileDescription": {"type": "object", "properties": {"contentType": {"type": "string", "description": "Тип файла"}, "fileName": {"type": "string", "data": {"type": "string", "format": "byte"}}, "description": "Файл запроса на сертификат"}, "EditSubjectRequestPublic": {"type": "object", "properties": {"id": {"type": "string", "description": "ID субъекта", "format": "uuid"}, "subjectName": {"type": "object", "additionalProperties": {"type": "array", "description": "Отличительное имя субъекта", "items": {"type": "string", "description": "Отличительное имя субъекта"}}, "description": "Отличительное имя субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя субъекта"}}, "description": "Дополнительное имя субъекта"}}, "description": "Запрос: создание (обновление) субъекта"}, "ItemResponseSubjectResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/SubjectResponsePublic"}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, "ResourceItemResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "ID ресурсной системы", "format": "uuid"}, "commonName": {"type": "string", "description": "Общее наименование (CN) ресурсной системы"}, "distinguishedName": {"type": "string", "description": "Отличающиеся имя ресурсной системы"}}, "description": "Ответ: ресурсная система (сокращенный)"}, "SubjectNameFieldItemResponsePublic": {"type": "object", "properties": {"values": {"type": "array", "description": "Значения атрибута субъекта", "items": {"type": "string", "description": "Значения атрибута субъекта"}}, "editable": {"type": "boolean", "description": "Флаг: редактируемый атрибут"}, "description": "Ответ: данные атрибута субъекта"}, "SubjectResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "ID субъекта", "format": "uuid"}, "commonName": {"type": "string", "description": "Общее наименование (CN) субъекта"}, "distinguishedName": {"type": "string", "description": "Отличающееся наименование (DN) субъекта"}, "resource": {"$ref": "#/components/schemas/ResourceItemResponsePublic"}, "subjectName": {"type": "object", "additionalProperties": {"$ref": "#/components/schemas/SubjectNameFieldItemResponsePublic"}, "description": "Отличительное имя субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"$ref": "#/components/schemas/SubjectNameFieldItemResponsePublic"}, "description": "Дополнительное имя субъекта"}, "isConnected": {"type": "boolean", "description": "Флаг: подключение субъекта к ресурсной системе"}, "isBlocked": {"type": "boolean", "description": "Флаг: блокировка субъекта в ресурсной системе"}, "modify": {"type": "string", "description": "Дата и время обновления в ресурсной системе", "format": "date-time"}, "certificatesCount": {"type": "integer", "description": "Количество сертификатов", "format": "int64"}, "updated": {"type": "string", "description": "Дата и время обновления", "format": "date-time"}, "created": {"type": "string", "description": "Дата и время создания", "format": "date-time"}}, "description": "Ответ: субъект"}, "EditSubjectByCsrRequestPublicMultipartFile": {"required": ["request"], "type": "object", "properties": {"id": {"type": "string", "description": "ID субъекта", "format": "uuid"}, "request": {"type": "string", "format": "binary"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя субъекта"}}, "description": "Дополнительное имя субъекта"}}, "description": "Запрос: создание (обновление) субъекта по запросу #PKCS10"}, "EditSubjectByCsrRequestPublicFileDescription": {"required": ["request"], "type": "object", "properties": {"id": {"type": "string", "description": "ID субъекта", "format": "uuid"}, "request": {"$ref": "#/components/schemas/FileDescription"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя субъекта"}}, "description": "Дополнительное имя субъекта"}}, "description": "Запрос: создание

```

(обновление) субъекта по запросу

```
#PKCS10"}, "CreateSyslogV2aRequestPublic": {"required": ["host", "port", "protocol"], "type": "object",
"properties": {"host": {"type": "string", "description": "Хост"}, "port": {"maximum": 65535, "minimum": 0, "type": "integer", "description": "Порт", "format": "int32"}, "protocol": {"type": "string", "description": "Протокол", "enum": ["UNKNOWN", "UDP", "TCP"]}}, "description": "Запрос
создания/редактирования syslog
сервера"}, "ItemResponseSyslogV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/SyslogV2aResponsePublic"}},
"description": "Стандартизированный объект ответа сервиса, содержащий один единственный
элемент"}, "SyslogV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Идентификатор syslog
сервера", "format": "uuid"}, "host": {"type": "string", "description": "Хост"}, "port": {"type": "integer", "description": "Порт", "format": "int32"}, "protocol": {"type": "string", "description": "Протокол", "enum": ["UNKNOWN", "UDP", "TCP"]}, "active": {"type": "boolean"}}, "description": "Объект ответа
сервиса, содержащий данные syslog
сервера"}, "CreateSyslogRequestPublic": {"required": ["host", "port", "protocol"], "type": "object", "properties": {"host": {"type": "string", "description": "Хост"}, "port": {"maximum": 65535, "minimum": 0, "type": "integer", "description": "Порт", "format": "int32"}, "protocol": {"type": "string", "description": "Протокол", "enum": ["UNKNOWN", "UDP", "TCP"]}}, "description": "Запрос
создания/редактирования syslog
сервера"}, "ItemResponseSyslogResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/SyslogResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный
элемент"}, "SyslogResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Идентификатор syslog
сервера", "format": "uuid"}, "host": {"type": "string", "description": "Хост"}, "port": {"type": "integer", "description": "Порт", "format": "int32"}, "protocol": {"type": "string", "description": "Протокол", "enum": ["UNKNOWN", "UDP", "TCP"]}, "active": {"type": "boolean"}}, "description": "Объект ответа
сервиса, содержащий данные syslog
сервера"}, "RevokeCertificateV2aRequestPublic": {"type": "object", "properties": {"reason": {"type": "string", "description": "Коды словаря: причины отзыва", "enum": ["UNKNOWN", "UNSPECIFIED", "KEY_COMPROMISE", "CA_COMPROMISE", "AFFILIATION_CHANGED", "SUPERSEDED", "CESSATION_OF_OPERATION", "CERTIFICATE_HOLD", "REMOVE_FROM_CRL", "PRIVILEGE_WITHDRAWN", "AA_COMPROMISE"]}}, "description": "Запрос: Отзыв
сертификата"}, "RevokeCertificateRequestPublic": {"type": "object", "properties": {"reason": {"type": "string", "description": "Коды словаря: причины отзыва", "enum": ["UNKNOWN", "UNSPECIFIED", "KEY_COMPROMISE", "CA_COMPROMISE", "AFFILIATION_CHANGED", "SUPERSEDED", "CESSATION_OF_OPERATION", "CERTIFICATE_HOLD", "REMOVE_FROM_CRL", "PRIVILEGE_WITHDRAWN", "AA_COMPROMISE"]}}, "description": "Запрос: Отзыв
сертификата"}, "ItemResponseTokenV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/TokenV2aResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный
элемент"}, "TokenV2aResponsePublic": {"type": "object", "properties": {"token": {"type": "string", "description": "Токен пользователя"}, "refresh": {"type": "string", "description": "Токен обновления
пользователя"}}, "description": "Объект ответа сервиса: данные о
токене"}, "ItemResponseTokenResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/TokenResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный
элемент"}, "TokenResponsePublic": {"type": "object", "properties": {"token": {"type": "string", "description": "Токен пользователя"}, "refresh": {"type": "string", "description": "Токен обновления
пользователя"}}, "description": "Объект ответа сервиса: данные о
токене"}, "ParseContainerV2aRequestPublic": {"type": "object", "properties": {"container": {"$ref": "#/components/schemas/FileDescription"}, "password": {"type": "string", "description": "Пароль от
контейнера"}, "templateId": {"type": "string", "description": "Идентификатор
шаблона", "format": "uuid"}}, "description": "Запрос: расшифровка контейнера
сертификата"}, "ContainerDetailsV2aResponsePublic": {"type": "object", "properties": {"serialnumber": {"type": "string", "description": "Серийный номер
сертификата"}, "fingerprint": {"type": "string", "description": "Фингерпринт
сертификата"}, "name": {"type": "string", "description": "Имя сертификата (формируется из CN,
разделитель
\"_\" )"}, "issuerDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Iss
```

```

uer DN", "items": {"type": "string", "description": "Issuer DN"}, "description": "Issuer
DN", "subjectDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Subjec
t DN", "items": {"type": "string", "description": "Subject DN"}, "description": "Subject
DN", "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "S
ubject AltName", "items": {"type": "string", "description": "Subject
AltName"}}, "description": "Subject AltName", "validFrom": {"type": "string", "description": "Дата
начала действия сертификата", "format": "date-
time"}, "validTo": {"type": "string", "description": "Дата окончания действия
сертификата", "format": "date-
time"}, "privateKey": {"$ref": "#/components/schemas/FileDescription"}, "certificate": {"$ref": "#/c
omponents/schemas/FileDescription"}, "keyUsages": {"type": "array", "description": "Разрешенные
использования
ключа", "items": {"$ref": "#/components/schemas/KeyUsageDictResponseV2aPublic"}}, "extendedKeyUsag
es": {"type": "array", "description": "Разрешенные расширенные использования
ключа", "items": {"$ref": "#/components/schemas/ExtendedKeyUsageResponseV2aPublic"}}, "ca": {"type":
"boolean"}, "description": "Ответ сервиса: данные контейнера
сертификата", "ExtendedKeyUsageResponseV2aPublic": {"type": "object", "properties": {"id": {"type":
"string", "description": "Идентификатор расширенного использования
ключа", "format": "uuid"}, "oid": {"type": "string", "description": "Идентификатор OID расширенного
использования ключа"}, "value": {"type": "string", "description": "Наименование расширенного
использования ключа"}, "description": {"type": "string", "description": "Описание использования
ключа"}, "created": {"type": "string", "description": "Дата расширенного использования
ключа", "format": "date-time"}, "updated": {"type": "string", "description": "Дата обновления
расширенного использования ключа", "format": "date-
time"}, "default": {"type": "boolean"}}, "description": "Разрешенные расширенные использования
ключа", "ItemResponseContainerDetailsV2aResponsePublic": {"type": "object", "properties": {"status":
{"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/ContainerDetailsV2a
ResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один
единственный
элемент", "KeyUsageDictResponseV2aPublic": {"type": "object", "properties": {"id": {"type": "string"
, "format": "uuid"}, "code": {"type": "string", "description": "Перечисление использования
ключа", "enum": ["UNKNOWN»,
"DIGITAL_SIGNATURE", "NON_REPUDIATION", "KEY_ENCIPHERMENT", "DATA_ENCIPHERMENT", "KEY_AGREEMENT", "
KEY_CERT_SIGN", "CRL_SIGN", "ENCIPHER_ONLY", "DECIPHER_ONLY"]}, "description": {"type": "string"}, "v
alue": {"type": "string"}}, "description": "Разрешенные использования
ключа", "ParseCsrV2aRequestPublic": {"type": "object", "properties": {"request": {"$ref": "#/compone
nts/schemas/FileDescription"}, "subjectName": {"type": "object", "additionalProperties": {"type": "a
rray", "description": "Дополнительное имя
субъекта", "items": {"type": "string", "description": "Дополнительное имя
субъекта"}}, "description": "Дополнительное имя
субъекта", "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "descripti
on": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя
субъекта"}}, "description": "Дополнительное имя субъекта"}}, "description": "Запрос: парсинг
данных
запроса", "ContainerDetailsPkcs10V2aResponsePublic": {"type": "object", "properties": {"name": {"ty
pe": "string", "description": "Имя сертификата (формируется из CN, разделитель
\"_\" )"}, "subjectDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Su
bject DN", "items": {"type": "string", "description": "Subject DN"}, "description": "Subject
DN", "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "S
ubject AltName", "items": {"type": "string", "description": "Subject
AltName"}}, "description": "Subject
AltName", "keyBits": {"type": "integer", "description": "Количество бит в
ключе", "format": "int32"}, "keyAlgorithm": {"$ref": "#/components/schemas/EKeyAlgorithmV2aPublic"}
, "subjectKeyIdentifier": {"type": "string", "description": "Идентификатор ключа
субъекта"}, "description": "Ответ сервиса: данные контейнера
сертификата", "EKeyAlgorithmV2aPublic": {"type": "string", "description": "Перечисление алгоритмов
ключа сертификата", "enum": ["UNKNOWN», «UNKNOWN»,
"RSA", "ECDSA", "GOST_R_34_10_2012", "UNKNOWN"]}, "ItemResponseContainerDetailsPkcs10V2aResponsePu
blic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$re
f": "#/components/schemas/ContainerDetailsPkcs10V2aResponsePublic"}}, "description": "Стандартизи
рованный объект ответа сервиса, содержащий один единственный

```

```

элемент"}, "ParsePemV2aRequestPublic": {"type": "object", "properties": {"request": {"$ref": "#/components/schemas/FileDescription"}}, "description": "Запрос: парсинг данных сертификата"}, "CertificateChainV2aPublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Уникальный идентификатор сертификата"}, "format": "uuid", "name": {"type": "string", "description": "Имя сертификата (формируется из CN, разделитель \"_\" )"}, "subjectDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Subject DN", "items": {"type": "string", "description": "Subject DN"}}, "description": "Subject DN"}, "issuer": {"$ref": "#/components/schemas/CertificateChainV2aPublic"}}, "description": "ПОЖО: цепочка сертификатов"}, "CertificateDownloadActionsV2aPublic": {"type": "object", "properties": {"p12": {"type": "boolean", "description": "Флаг: выгрузка контейнера p12"}, "pem": {"type": "boolean", "description": "Флаг: выгрузка сертификата"}, "csr": {"type": "boolean", "description": "Флаг: выгрузка запроса сертификата"}}, "description": "ПОЖО: Действия выгрузки у сертификата"}, "CertificateV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Уникальный идентификатор сертификата"}, "format": "uuid", "chain": {"$ref": "#/components/schemas/CertificateChainV2aPublic"}, "serialnumber": {"type": "string", "description": "Серийный номер сертификата"}, "fingerprint": {"type": "string", "description": "Фингерпринт сертификата"}, "name": {"type": "string", "description": "Имя сертификата (формируется из CN, разделитель \"_\" )"}, "issuerId": {"type": "string", "description": "Уникальный идентификатор издающего сертификата"}, "format": "uuid", "issuerDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Issuer DN", "items": {"type": "string", "description": "Issuer DN"}}, "description": "Issuer DN"}, "subjectDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Subject DN", "items": {"type": "string", "description": "Subject DN"}}, "description": "Subject DN"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "Subject AltName", "items": {"type": "string", "description": "Subject AltName"}}, "description": "Subject AltName"}, "keyAlgorithm": {"$ref": "#/components/schemas/EKeyAlgorithmV2aPublic"}, "hashAlgorithm": {"type": "string", "description": "Перечисление Hash алгоритмов", "enum": ["UNKNOWN", "UNKNOWN", "SHA1", "SHA256", "SHA384", "SHA512", "SHA3_256", "SHA3_384", "SHA3_512", "RSASSA_PSS", "MD5", "GOST_R_34_11_2012", "UNKNOWN"]}, "keyBits": {"type": "integer", "description": "Размерность ключа", "format": "int32"}, "templateId": {"type": "string", "description": "Идентификатор шаблона, по которому выпущен сертификат", "format": "uuid"}, "templateName": {"type": "string", "description": "Название шаблона, по которому выпущен сертификат"}, "type": {"type": "string", "description": "Перечисление типов сертификата", "enum": ["UNKNOWN", "UNKNOWN", "CERTIFICATE", "ROOT_CA", "SUB_CA"]}, "validFrom": {"type": "string", "description": "Дата начала действия сертификата", "format": "date-time"}, "validTo": {"type": "string", "description": "Дата окончания действия сертификата", "format": "date-time"}, "status": {"type": "string", "description": "Перечисление: статус сертификата", "enum": ["UNKNOWN", "UNKNOWN", "ACTIVE", "HOLD", "REVOKE", "REQUEST"]}, "isValid": {"type": "boolean", "description": "Сертификат действителен"}, "isExpired": {"type": "boolean", "description": "Сертификат истек"}, "actions": {"$ref": "#/components/schemas/CertificateDownloadActionsV2aPublic"}, "publicKey": {"type": "string", "description": "Публичный ключ сертификата"}, "certificateType": {"type": "string", "description": "Тип сертификата"}, "version": {"type": "integer", "description": "Версия сертификата", "format": "int32"}, "subjectKeyIdentifier": {"type": "string", "description": "Идентификатор ключа сертификата"}, "authorityKeyIdentifier": {"type": "string", "description": "Идентификатор ключа издателя"}, "keyUsages": {"type": "array", "description": "Разрешенные использования ключа", "items": {"$ref": "#/components/schemas/KeyUsageDictResponseV2aPublic"}}, "extendedKeyUsages": {"type": "array", "description": "Разрешенные расширенные использования ключа", "items": {"$ref": "#/components/schemas/ExtendedKeyUsageResponseV2aPublic"}}, "ca": {"type": "boolean", "description": "Флаг: сертификат ЦС"}, "revocation": {"$ref": "#/components/schemas/RevocationInfoV2aPublic"}, "aiaUrls": {"type": "array", "description": "Точки распространения AIA", "items": {"type": "string", "description": "Точки

```

```

распространения AIA"}}, "ocspUrls": {"type": "array", "description": "URL сервисов
OCSP", "items": {"type": "string", "description": "URL сервисов
OCSP"}}, "crlUrls": {"type": "array", "description": "Точки распространения
CRL", "items": {"type": "string", "description": "Точки распространения
CRL"}}, "deltaCrlUrls": {"type": "array", "description": "Точки распространения Delta
CRL", "items": {"type": "string", "description": "Точки распространения Delta
CRL"}}, "userId": {"type": "string", "description": "Идентификатор
пользователя", "format": "uuid"}, "subjectId": {"type": "string", "description": "Идентификатор
субъекта", "format": "uuid"}, "created": {"type": "string", "description": "Дата создания
шаблона", "format": "date-time"}, "updated": {"type": "string", "description": "Дата обновления
шаблона", "format": "date-time"}, "description": "Ответ сервиса: данные
сертификата", "ItemResponseCertificateV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/CertificateV2aResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один
единственный
элемент"}, "RevocationInfoV2aPublic": {"type": "object", "properties": {"date": {"type": "string", "description": "Дата отзыва", "format": "date-time"}, "number": {"type": "integer", "description": "Номер
причины отзыва", "format": "int32"}, "value": {"type": "string", "description": "Значение причины
отзыва"}, "EnrollCsrV2aRequestPublicMultipartFile": {"type": "object", "properties": {"templateId": {"type": "string", "description": "Идентификатор
шаблона", "format": "uuid"}, "request": {"type": "string", "format": "binary"}, "subjectName": {"type": "object", "additionalProperties": {"type": "array", "description": "Имя
субъекта", "items": {"type": "string", "description": "Имя субъекта"}}, "description": "Имя
субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя
субъекта"}}, "description": "Дополнительное имя субъекта"}, "description": "Запрос: выпуск
сертификата по
запросу"}, "ItemResponseValidateV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/ValidateV2aResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный
элемент"}, "SubjectAltNameErrorV2aPublic": {"type": "object", "properties": {"fieldName": {"type": "string", "description": "Перечисление полей Subject AltName", "enum": ["UNKNOWN", "RFC822NAME", "DNS_NAME", "IPADDRESS", "DIRECTORY_NAME", "UNIFORM_RESOURCE_ID", "REGISTERED_ID", "MS_UPN", "MS_GUID", "KRB5PRINCIPAL", "PERMANENT_IDENTIFIER", "XMPP_ADDR", "SRV_NAME", "SUBJECT_IDENTIFICATION_METHOD"]}, "fieldValue": {"type": "string"}, "required": {"type": "boolean"}, "additional": {"type": "boolean"}, "valid": {"type": "boolean"}, "message": {"type": "string"}}, "SubjectNameErrorV2aPublic": {"type": "object", "properties": {"fieldName": {"type": "string", "description": "Перечислени
е полей Subject DN", "enum": ["UNKNOWN", "EMAILADDRESS", "CN", "UID", "SERIALNUMBER", "OU", "O", "L", "ST", "DC", "C", "T", "SURNAME", "STREET", "INITIALS", "GIVENNAME", "UNSTRUCTUREDADDRESS", "UNSTRUCTUREDNAME", "POSTALCODE", "BUSINESSCATEGORY", "TELEPHONENUMBER", "PSEUDONYM", "POSTALADDRESS", "NAME", "DN", "DESCRIPTION", "INN", "OGRN", "OGRNIP", "SNILS", "INNLE"]}, "fieldValue": {"type": "string"}, "required": {"type": "boolean"}, "additional": {"type": "boolean"}, "valid": {"type": "boolean"}, "message": {"type": "string"}}, "ValidateV2aResponsePublic": {"type": "object", "properties": {"name": {"type": "string"}, "templateName": {"type": "string"}, "templateId": {"type": "string", "format": "uuid"}, "subjectId": {"type": "string", "format": "uuid"}, "valid": {"type": "boolean"}, "subjectNames": {"type": "array", "items": {"$ref": "#/components/schemas/SubjectNameErrorV2aPublic"}}, "subjectAltNames": {"type": "array", "items": {"$ref": "#/components/schemas/SubjectAltNameErrorV2aPublic"}}, "description": "Полезная
нагрузка"}, "EnrollCsrV2aRequestPublicFileDescription": {"type": "object", "properties": {"templateId": {"type": "string", "description": "Идентификатор
шаблона", "format": "uuid"}, "request": {"$ref": "#/components/schemas/FileDescription"}, "subjectName": {"type": "object", "additionalProperties": {"type": "array", "description": "Имя
субъекта", "items": {"type": "string", "description": "Имя субъекта"}}, "description": "Имя
субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя
субъекта"}}, "description": "Дополнительное имя субъекта"}, "description": "Запрос: выпуск
сертификата по
запросу"}, "CreateCertificateV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Идентификатор
сертификата", "format": "uuid"}, "downloadActions": {"$ref": "#/components/schemas/CertificateDownl

```

```
oadActionsV2aPublic"},"fingerprint":{"type":"string","description":"Фингерпринт
сертификата"},"serialNumber":{"type":"string","description":"Серийный номер
сертификата"},"templateId":{"type":"string","description":"Идентификатор шаблона
сертификата"},"format":"uuid"},"templateName":{"type":"string","description":"Название шаблона
сертификата"},"name":{"type":"string","description":"Название
сертификата"},"issuerDN":{"type":"object","additionalProperties":{"type":"array","description":
"Issuer DN сертификата"},"items":{"type":"string","description":"Issuer DN
сертификата"},"description":"Issuer DN
сертификата"},"subjectDN":{"type":"object","additionalProperties":{"type":"array","description":
"Subject DN сертификата"},"items":{"type":"string","description":"Subject DN
сертификата"},"description":"Subject DN
сертификата"},"subjectAltName":{"type":"object","additionalProperties":{"type":"array","descri
ption":"Subject AltName сертификата"},"items":{"type":"string","description":"Subject AltName
сертификата"},"description":"Subject AltName
сертификата"},"validFrom":{"type":"string","description":"Начало действия
сертификата"},"format":"date-time"},"validTo":{"type":"string","description":"Окончание
действия сертификата"},"format":"date-time"},"description":"Ответ сервиса: сертификат
создан"},"ItemResponseCreateCertificateV2aResponsePublic":{"type":"object","properties":{"stat
us":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/CreateCertificate
V2aResponsePublic"},"description":"Стандартизированный объект ответа сервиса, содержащий один
единственный
элемент"},"EnrollCertificateV2aRequestPublic":{"type":"object","properties":{"templateId":{"ty
pe":"string","description":"Идентификатор
шаблона"},"format":"uuid"},"subjectDN":{"type":"object","additionalProperties":{"type":"array",
"description":"Отличительное имя
субъекта"},"items":{"type":"string","description":"Отличительное имя
субъекта"},"description":"Отличительное имя
субъекта"},"subjectAltName":{"type":"object","additionalProperties":{"type":"array","descripti
on":"Дополнительное имя субъекта"},"items":{"type":"string","description":"Дополнительное имя
субъекта"},"description":"Дополнительное имя
субъекта"},"keyBits":{"type":"integer","description":"Длина
ключа"},"format":"int32"},"keyAlgorithm":{"$ref":"#/components/schemas/EKeyAlgorithmV2aPublic"}
,"password":{"type":"string","description":"Пароль"},"description":"Запрос: выпуск
сертификата с закрытым
ключом"},"EnrollCsrRequestPublicMultipartFile":{"type":"object","properties":{"templateId":{"t
ype":"string","description":"Идентификатор
шаблона"},"format":"uuid"},"request":{"type":"string","format":"binary"},"subjectAltName":{"typ
e":"object","additionalProperties":{"type":"array","description":"Дополнительное имя
субъекта"},"items":{"type":"string","description":"Дополнительное имя
субъекта"},"description":"Дополнительное имя субъекта"},"description":"Запрос: выпуск
сертификата по
запросу"},"ItemResponseValidateResponsePublic":{"type":"object","properties":{"status":{"type":
"integer","format":"int32"},"data":{"$ref":"#/components/schemas/ValidateResponsePublic"},"d
escription":"Стандартизированный объект ответа сервиса, содержащий один единственный
элемент"},"SubjectAltNameErrorPublic":{"type":"object","properties":{"fieldName":{"type":"stri
ng","description":"Перечисление полей Subject AltName"},"enum":["UNKNOWN",
"RFC822NAME","DNS_NAME","IPADDRESS","DIRECTORY_NAME","UNIFORM_RESOURCE_ID","REGISTERED_ID","MS
_UPN","MS_GUID","KRB5PRINCIPAL","PERMANENT_IDENTIFIER","XMPP_ADDR","SRV_NAME","SUBJECT_IDENTIF
ICATION_METHOD"]},"fieldValue":{"type":"string"},"required":{"type":"boolean"},"additional":{"t
ype":"boolean"},"valid":{"type":"boolean"},"message":{"type":"string"}}},"SubjectNameErrorPub
lic":{"type":"object","properties":{"fieldName":{"type":"string","description":"Перечисление
полей Subject DN"},"enum":["UNKNOWN",
"EMAILADDRESS","CN","UID","SERIALNUMBER","OU","O","L","ST","DC","C","T","SURNAME","STREET","IN
ITIALS","GIVENNAME","UNSTRUCTUREDADDRESS","UNSTRUCTUREDNAME","POSTALCODE","BUSINESSCATEGORY","
TELEPHONENUMBER","PSEUDONYM","POSTALADDRESS","NAME","DN","DESCRIPTION","INN","OGRN","OGRNIP","
SNILS","INNLE"]},"fieldValue":{"type":"string"},"required":{"type":"boolean"},"additional":{"t
ype":"boolean"},"valid":{"type":"boolean"},"message":{"type":"string"}}},"ValidateResponsePubl
ic":{"type":"object","properties":{"name":{"type":"string"},"templateName":{"type":"string"},"t
emplateId":{"type":"string","format":"uuid"},"subjectId":{"type":"string","format":"uuid"},"v
alid":{"type":"boolean"},"subjectNames":{"type":"array","items":{"$ref":"#/components/schemas/
SubjectNameErrorPublic"},"subjectAltNames":{"type":"array","items":{"$ref":"#/components/sche
```

```

mas/SubjectAltNameErrorPublic"}}, {"description": "Полезная
нагрузка"}, {"EnrollCsrRequestPublicFileDescription": {"type": "object", "properties": {"templateId":
{"type": "string", "description": "Идентификатор
шаблона", "format": "uuid"}, "request": {"$ref": "#/components/schemas/FileDescription"}, "subjectAl
tName": {"type": "object", "additionalProperties": {"type": "array", "description": "Дополнительное
имя субъекта", "items": {"type": "string", "description": "Дополнительное имя
субъекта"}}, "description": "Дополнительное имя субъекта"}}, {"description": "Запрос: выпуск
сертификата по запросу"}, {"EKeyAlgorithmPublic": {"type": "string", "description": "Перечисление
алгоритмов ключа сертификата", "enum": ["UNKNOWN",
"RSA", "ECDSA", "GOST_R_34_10_2012", "UNKNOWN"]}, {"EnrollCertificateRequestPublic": {"type": "object
", "properties": {"templateId": {"type": "string", "description": "Идентификатор
шаблона", "format": "uuid"}, "subjectDN": {"type": "object", "additionalProperties": {"type": "array",
"description": "Отличительное имя
субъекта", "items": {"type": "string", "description": "Отличительное имя
субъекта"}}, "description": "Отличительное имя
субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "descripti
on": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя
субъекта"}}, "description": "Дополнительное имя
субъекта"}, "keyBits": {"type": "integer", "description": "Длина
ключа", "format": "int32"}, "keyAlgorithm": {"$ref": "#/components/schemas/EKeyAlgorithmPublic"}, "p
assword": {"type": "string", "description": "Пароль"}}, {"description": "Запрос: выпуск сертификата с
закрытым
ключом"}, {"CertificateDownloadActionsPublic": {"type": "object", "properties": {"p12": {"type": "bool
ean", "description": "Флаг: выгрузка контейнера
p12"}, "pem": {"type": "boolean", "description": "Флаг: выгрузка
сертификата"}, "csr": {"type": "boolean", "description": "Флаг: выгрузка запроса
сертификата"}}, {"description": "POJO: Действия выгрузки у
сертификата"}, {"CreateCertificateResponsePublic": {"type": "object", "properties": {"id": {"type": "s
tring", "description": "Идентификатор
сертификата", "format": "uuid"}, "downloadActions": {"$ref": "#/components/schemas/CertificateDownl
oadActionsPublic"}, "fingerprint": {"type": "string", "description": "Фингерпринт
сертификата"}, "serialNumber": {"type": "string", "description": "Серийный номер
сертификата"}, "templateId": {"type": "string", "description": "Идентификатор шаблона
сертификата", "format": "uuid"}, "templateName": {"type": "string", "description": "Название шаблона
сертификата"}, "name": {"type": "string", "description": "Название
сертификата"}, "issuerDN": {"type": "object", "additionalProperties": {"type": "array", "description"
: "Issuer DN сертификата", "items": {"type": "string", "description": "Issuer DN
сертификата"}}, {"description": "Issuer DN
сертификата"}, "subjectDN": {"type": "object", "additionalProperties": {"type": "array", "description
": "Subject DN сертификата", "items": {"type": "string", "description": "Subject DN
сертификата"}}, {"description": "Subject DN
сертификата"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "descri
ption": "Subject AltName сертификата", "items": {"type": "string", "description": "Subject AltName
сертификата"}}, {"description": "Subject AltName
сертификата"}, "validFrom": {"type": "string", "description": "Начало действия
сертификата", "format": "date-time"}, "validTo": {"type": "string", "description": "Окончание
действия сертификата", "format": "date-time"}}, {"description": "Ответ сервиса: сертификат
создан"}, {"ItemResponseCreateCertificateResponsePublic": {"type": "object", "properties": {"status"
: {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/CreateCertificateRes
ponsePublic"}}, {"description": "Стандартизированный объект ответа сервиса, содержащий один
единственный
элемент"}, {"CollectionResponseTemplateItemV2aResponsePublic": {"type": "object", "properties": {"st
atus": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataTemplateIte
mV2aResponsePublic"}}, {"description": "Стандартизированный объект ответа сервиса, содержащий
коллекцию
элементов"}, {"DataTemplateItemV2aResponsePublic": {"type": "object", "properties": {"range": {"$ref"
: "#/components/schemas/Range"}, "items": {"type": "array", "description": " Коллекция
элементов", "items": {"$ref": "#/components/schemas/TemplateItemV2aResponsePublic"}}, {"descriptio
n": "Полезная нагрузка с коллекцией
элементов"}, {"Range": {"type": "object", "properties": {"count": {"type": "integer", "description": "
Всего элементов", "format": "int64"}, "offset": {"type": "integer", "description": " Количество

```

пропущенных элементов", "format": "int64"}, "limit": {"type": "integer", "description": "Ограничение количества элементов в выборке", "format": "int32"}, "totalPages": {"type": "integer", "description": "Общее число страниц", "format": "int32"}, "pageNumber": {"type": "integer", "description": "Номер страницы выборки", "format": "int32"}}, "description": "Данные по выборке", "TemplateItemV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Идентификатор шаблона", "format": "uuid"}, "name": {"type": "string", "description": "Наименование шаблона"}, "endEntityType": {"type": "string", "description": "Перечисление типов сертификата", "enum": ["UNKNOWN", "ROOT_CA", "SUB_CA", "USER", "DEVICE"]}, "type": {"type": "string", "description": "Тип шаблона", "enum": ["UNKNOWN", "EMBEDDED", "CLONED", "IMPORTED"]}, "certificateCount": {"type": "integer", "description": "Количество выпущенных сертификатов", "format": "int64"}, "removed": {"type": "boolean", "description": "Удаленный шаблон"}, "created": {"type": "string", "description": "Дата создания шаблона", "format": "date-time"}, "updated": {"type": "string", "description": "Дата обновления шаблона", "format": "date-time"}}, "description": "Объект ответа сервиса, содержащий данные элемента коллекции шаблонов", "AecaCertificatePoliciesV2aPublic": {"type": "object", "properties": {"critical": {"type": "boolean", "description": "Считать критическим"}, "values": {"type": "array", "description": "Перечень политик", "items": {"type": "string", "description": "Перечень политик"}}, "description": "Класс политик сертификата", "AecaExtendedKeyUsageV2aPublic": {"type": "object", "properties": {"critical": {"type": "boolean", "description": "Считать критическим"}, "values": {"type": "array", "description": "Значения (oids)", "items": {"type": "string", "description": "Значения (oids)"}}, "description": "Класс расширенного использования ключа", "AecaKeyAlgorithmV2aPublic": {"type": "object", "properties": {"use": {"type": "boolean", "description": "Используется ли"}, "minLength": {"type": "integer", "description": "Минимальная длина ключа", "format": "int32"}, "lengths": {"type": "array", "description": "Длины алгоритма с учетом минимальной длины", "items": {"type": "integer", "description": "Длины алгоритма с учетом минимальной длины", "format": "int32"}}, "description": "Объект сервиса, содержащего данные алгоритма ключа сертификата", "AecaKeyUsageV2aPublic": {"type": "object", "properties": {"critical": {"type": "boolean", "description": "Считать критическим"}, "values": {"type": "array", "description": "Значения", "items": {"type": "string", "description": "Перечисление использования ключа", "enum": ["UNKNOWN", "DIGITAL_SIGNATURE", "NON_REPUDIATION", "KEY_ENCIPHERMENT", "DATA_ENCIPHERMENT", "KEY_AGREEMENT", "KEY_CERT_SIGN", "CRL_SIGN", "ENCIPHER_ONLY", "DECIPHER_ONLY"]}}, "description": "Класс использования ключа", "ItemResponseTemplateV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"type": "string", "format": "url", "description": "\$ref:#/components/schemas/TemplateV2aResponsePublic"}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный элемент", "SubjectAltNameFieldV2aPublic": {"required": ["code", "generalName", "required"], "type": "object", "properties": {"index": {"type": "integer", "description": "Индекс поля имени субъекта", "format": "int32"}, "name": {"type": "string", "description": "Полное имя поля имени субъекта"}, "description": {"type": "string", "description": "Описание поля имени субъекта"}, "required": {"type": "boolean", "description": "Обязательный ли"}, "validation": {"type": "boolean", "description": "Используется ли валидация"}, "modifiable": {"type": "boolean", "description": "Редактируемый ли"}, "regex": {"type": "string", "description": "Регулярное выражение"}, "defaultValue": {"type": "string", "description": "Значение по умолчанию"}, "alert": {"type": "string", "description": "Предупреждающее сообщение поля имени субъекта"}, "code": {"type": "string", "description": "Перечисление полей Subject AltName", "enum": ["UNKNOWN", "RFC822NAME", "DNS_NAME", "IPADDRESS", "DIRECTORY_NAME", "UNIFORM_RESOURCE_ID", "REGISTERED_ID", "MS_UPN", "MS_GUID", "KRB5PRINCIPAL", "PERMANENT_IDENTIFIER", "XMPP_ADDR", "SRV_NAME", "SUBJECT_IDENTIFICATION_METHOD"]}, "generalName": {"type": "integer", "description": "Идентификатор имени в RFC", "format": "int32"}, "oid": {"type": "string", "description": "OID в RFC"}}, "description": "Класс поля дополнительного имени субъекта", "SubjectNameFieldV2aPublic": {"required": ["code", "required"], "type": "object", "proper

```

ties":{"index":{"type":"integer","description":"Индекс поля имени
субъекта","format":"int32"},"name":{"type":"string","description":"Полное имя поля имени
субъекта"},"description":{"type":"string","description":"Описание поля имени
субъекта"},"required":{"type":"boolean","description":"Обязательный
ли"},"validation":{"type":"boolean","description":"Используется ли
валидация"},"modifiable":{"type":"boolean","description":"Редактируемый
ли"},"regex":{"type":"string","description":"Регулярное
выражение"},"defaultValue":{"type":"string","description":"Значение по
умолчанию"},"alert":{"type":"string","description":"Предупреждающее сообщение поля имени
субъекта"},"code":{"type":"string","description":"Перечисление полей Subject
DN"},"enum":["«UNKNOWN»",
"EMAILADDRESS","CN","UID","SERIALNUMBER","OU","O","L","ST","DC","C","T","SURNAME","STREET","IN
ITIALS","GIVENNAME","UNSTRUCTUREDADDRESS","UNSTRUCTUREDNAME","POSTALCODE","BUSINESSCATEGORY","
TELEPHONENUMBER","PSEUDONYM","POSTALADDRESS","NAME","DN","DESCRIPTION","INN","OGRN","OGRNIP","
SNILS","INNLE"]},"description":"Класс поля отличительного имени
субъекта"},"TemplateV2aResponsePublic":{"type":"object","properties":{"id":{"type":"string","d
escription":"Идентификатор
шаблона"},"format":"uuid"},"name":{"type":"string","description":"Наименование
шаблона"},"validity":{"type":"integer","description":"Срок действия
сертификата"},"format":"int64"},"endEntityType":{"type":"string","description":"Перечисление
типов сертификата"},"enum":["«UNKNOWN»", «ROOT_CA», «SUB_CA», «USER»,
«DEVICE»]},"type":{"type":"string","description":"Тип шаблона"},"enum":["«UNKNOWN»",
"EMBEDDED","CLONED","IMPORTED"]},"removed":{"type":"boolean","description":"Удаленный
шаблон"},"rsa":{"$ref":"#/components/schemas/AecaKeyAlgorithmV2aPublic"},"ecdsa":{"$ref":"#/co
mponents/schemas/AecaKeyAlgorithmV2aPublic"},"gost":{"$ref":"#/components/schemas/AecaKeyAlgor
ithmV2aPublic"},"keyUsage":{"$ref":"#/components/schemas/AecaKeyUsageV2aPublic"},"extendedKeyU
sage":{"$ref":"#/components/schemas/AecaExtendedKeyUsageV2aPublic"},"policies":{"$ref":"#/comp
onents/schemas/AecaCertificatePoliciesV2aPublic"},"subjectDN":{"type":"array","description":"О
тличительное имя
субъекта"},"items":{"$ref":"#/components/schemas/SubjectNameFieldV2aPublic"}}, "subjectAltName":
{"type":"array","description":"Дополнительное имя
субъекта"},"items":{"$ref":"#/components/schemas/SubjectAltNameFieldV2aPublic"}}, "created":{"ty
pe":"string","description":"Дата создания шаблона","format":"date-
time"},"updated":{"type":"string","description":"Дата обновления шаблона","format":"date-
time"}}, "description":"Объект ответа сервиса, содержащий данные
шаблона"},"CollectionResponseTemplateItemResponsePublic":{"type":"object","properties":{"statu
s":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataTemplateItemRe
sponsePublic"}}, "description":"Стандартизированный объект ответа сервиса, содержащий коллекцию
элементов"},"DataTemplateItemResponsePublic":{"type":"object","properties":{"range":{"$ref":"#
/components/schemas/Range"},"items":{"type":"array","description":" Коллекция
элементов"},"items":{"$ref":"#/components/schemas/TemplateItemResponsePublic"}}, "description":
"Полезная нагрузка с коллекцией
элементов"},"TemplateItemResponsePublic":{"type":"object","properties":{"id":{"type":"string",
"description":"Идентификатор
шаблона"},"format":"uuid"},"name":{"type":"string","description":"Наименование
шаблона"},"endEntityType":{"type":"string","description":"Перечисление типов
сертификата"},"enum":["«UNKNOWN»", "USER","DEVICE",
"ROOT_CA","SUB_CA"]},"type":{"type":"string","description":"Тип шаблона"},"enum":["«UNKNOWN»",
"EMBEDDED","CLONED","IMPORTED"]},"certificateCount":{"type":"integer","description":"Количеств
о выпущенных
сертификатов"},"format":"int64"},"removed":{"type":"boolean","description":"Удаленный
шаблон"},"created":{"type":"string","description":"Дата создания шаблона","format":"date-
time"},"updated":{"type":"string","description":"Дата обновления шаблона","format":"date-
time"}}, "description":"Объект ответа сервиса, содержащий данные элемента коллекции
шаблонов"},"AecaCertificatePoliciesPublic":{"type":"object","properties":{"critical":{"type":"
boolean","description":"Считать критическим"},"values":{"type":"array","description":"Перечень
политик"},"items":{"type":"string","description":"Перечень политик"}}, "description":"Класс
политик
сертификата"},"AecaExtendedKeyUsagePublic":{"type":"object","properties":{"critical":{"type":"
boolean","description":"Считать критическим"},"values":{"type":"array","description":"Значения
(oids)","items":{"type":"string","description":"Значения (oids)"}}, "description":" Класс

```

расширенного использования

```

ключа"}, "AecaKeyAlgorithmPublic": {"type": "object", "properties": {"use": {"type": "boolean", "description": "Используется ли"}, "minLength": {"type": "integer", "description": "Минимальная длина ключа", "format": "int32"}, "lengths": {"type": "array", "description": "Длины алгоритма с учетом минимальной длины", "items": {"type": "integer", "description": "Длины алгоритма с учетом минимальной длины", "format": "int32"}}}, "description": "Объект сервиса, содержащего данные алгоритма ключа сертификата"}, "AecaKeyUsagePublic": {"type": "object", "properties": {"critical": {"type": "boolean", "description": "Считать критическим"}, "values": {"type": "array", "description": "Значения", "items": {"type": "string", "description": "Перечисление использования ключа", "enum": ["UNKNOWN", "DIGITAL_SIGNATURE", "NON_REPUDIATION", "KEY_ENCIPHERMENT", "DATA_ENCIPHERMENT", "KEY_AGREEMENT", "KEY_CERT_SIGN", "CRL_SIGN", "ENCIPHER_ONLY", "DECIPHER_ONLY"]}}}, "description": "Класс использования ключа"}, "ItemResponseTemplateResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/TemplateResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, "SubjectAltNameFieldPublic": {"required": ["code", "generalName", "required"], "type": "object", "properties": {"index": {"type": "integer", "description": "Индекс поля имени субъекта", "format": "int32"}, "name": {"type": "string", "description": "Полное имя поля имени субъекта"}, "description": {"type": "string", "description": "Описание поля имени субъекта"}, "required": {"type": "boolean", "description": "Обязательный ли"}, "validation": {"type": "boolean", "description": "Используется ли валидация"}, "modifiable": {"type": "boolean", "description": "Редактируемый ли"}, "regex": {"type": "string", "description": "Регулярное выражение"}, "defaultValue": {"type": "string", "description": "Значение по умолчанию"}, "alert": {"type": "string", "description": "Предупреждающее сообщение поля имени субъекта"}, "code": {"type": "string", "description": "Перечисление полей Subject AltName", "enum": ["UNKNOWN", "RFC822NAME", "DNS_NAME", "IPADDRESS", "DIRECTORY_NAME", "UNIFORM_RESOURCE_ID", "REGISTERED_ID", "MS_UPN", "MS_GUID", "KRB5PRINCIPAL", "PERMANENT_IDENTIFIER", "XMPP_ADDR", "SRV_NAME", "SUBJECT_IDENTIFICATION_METHOD"]}, "generalName": {"type": "integer", "description": "Идентификатор имени в RFC", "format": "int32"}, "oid": {"type": "string", "description": "OID в RFC"}}, "description": "Класс поля дополнительного имени субъекта"}, "SubjectNameFieldPublic": {"required": ["code", "required"], "type": "object", "properties": {"index": {"type": "integer", "description": "Индекс поля имени субъекта", "format": "int32"}, "name": {"type": "string", "description": "Полное имя поля имени субъекта"}, "description": {"type": "string", "description": "Описание поля имени субъекта"}, "required": {"type": "boolean", "description": "Обязательный ли"}, "validation": {"type": "boolean", "description": "Используется ли валидация"}, "modifiable": {"type": "boolean", "description": "Редактируемый ли"}, "regex": {"type": "string", "description": "Регулярное выражение"}, "defaultValue": {"type": "string", "description": "Значение по умолчанию"}, "alert": {"type": "string", "description": "Предупреждающее сообщение поля имени субъекта"}, "code": {"type": "string", "description": "Перечисление полей Subject DN", "enum": ["UNKNOWN", "EMAILADDRESS", "CN", "UID", "SERIALNUMBER", "OU", "O", "L", "ST", "DC", "C", "T", "SURNAME", "STREET", "INITIALS", "GIVENNAME", "UNSTRUCTUREDADDRESS", "UNSTRUCTUREDNAME", "POSTALCODE", "BUSINESSCATEGORY", "TELEPHONENUMBER", "PSEUDONYM", "POSTALADDRESS", "NAME", "DN", "DESCRIPTION", "INN", "OGRN", "OGRNIP", "SNILS", "INNLE"]}}}, "description": "Класс поля отличительного имени субъекта"}, "TemplateResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Идентификатор шаблона", "format": "uuid"}, "name": {"type": "string", "description": "Наименование шаблона"}, "validity": {"type": "integer", "description": "Срок действия сертификата", "format": "int64"}, "endEntityType": {"type": "string", "description": "Перечисление типов сертификата", "enum": ["UNKNOWN", "USER", "DEVICE", "ROOT_CA", "SUB_CA"]}, "type": {"type": "string", "description": "Тип шаблона", "enum": ["UNKNOWN", "EMBEDDED", "CLONED", "IMPORTED"]}, "removed": {"type": "boolean", "description": "Удаленный шаблон"}, "rsa": {"$ref": "#/components/schemas/AecaKeyAlgorithmPublic"}, "ecdsa": {"$ref": "#/components/schemas/AecaKeyAlgorithmPublic"}, "gost": {"$ref": "#/components/schemas/AecaKeyAlgorithmPublic"}, "keyUsage": {"$ref": "#/components/schemas/AecaKeyUsagePublic"}, "extendedKeyUsage": {"$ref":

```

```

":{"#/components/schemas/AecaExtendedKeyUsagePublic"},"policies":{"$ref":"#/components/schemas/AecaCertificatePoliciesPublic"},"subjectDN":{"type":"array","description":"Отличительное имя субъекта","items":{"$ref":"#/components/schemas/SubjectNameFieldPublic"}}, "subjectAltName":{"type":"array","description":"Дополнительное имя субъекта","items":{"$ref":"#/components/schemas/SubjectAltNameFieldPublic"}}, "created":{"type":"string","description":"Дата создания шаблона","format":"date-time"},"updated":{"type":"string","description":"Дата обновления шаблона","format":"date-time"}}, "description":"Объект ответа сервиса, содержащий данные шаблона"},"CollectionResponseSubjectV2aResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataSubjectV2aResponsePublic"}}, "description":"Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"},"DataSubjectV2aResponsePublic":{"type":"object","properties":{"range":{"$ref":"#/components/schemas/Range"},"items":{"type":"array","description":"Коллекция элементов","items":{"$ref":"#/components/schemas/SubjectV2aResponsePublic"}}}, "description":"Полезная нагрузка с коллекцией элементов"},"CollectionResponseUUID":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataUUID"}}, "description":"Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"},"DataUUID":{"type":"object","properties":{"range":{"$ref":"#/components/schemas/Range"},"items":{"type":"array","description":"Коллекция элементов","items":{"type":"string","description":"Коллекция элементов","format":"uuid"}}}, "description":"Полезная нагрузка с коллекцией элементов"},"CollectionResponseSecurityGroupV2aResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataSecurityGroupV2aResponsePublic"}}, "description":"Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"},"DataSecurityGroupV2aResponsePublic":{"type":"object","properties":{"range":{"$ref":"#/components/schemas/Range"},"items":{"type":"array","description":"Коллекция элементов","items":{"$ref":"#/components/schemas/SecurityGroupV2aResponsePublic"}}}, "description":"Полезная нагрузка с коллекцией элементов"},"SecurityGroupV2aResponsePublic":{"type":"object","properties":{"id":{"type":"string","description":"ID группы безопасности","format":"uuid"},"commonName":{"type":"string","description":"Общее наименование (CN) группы безопасности"},"distinguishedName":{"type":"string","description":"Отличающееся наименование (DN) группы безопасности"},"resource":{"$ref":"#/components/schemas/ResourceItemV2aResponsePublic"},"isActive":{"type":"boolean","description":"Флаг: активная для назначения прав группа безопасности"},"modify":{"type":"string","description":"Дата и время обновления в ресурсной системе","format":"date-time"},"updated":{"type":"string","description":"Дата и время обновления","format":"date-time"},"created":{"type":"string","description":"Дата и время создания","format":"date-time"}}, "description":"Ответ: группа безопасности"},"ItemResponseSecurityGroupV2aResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/SecurityGroupV2aResponsePublic"}}, "description":"Стандартизированный объект ответа сервиса, содержащий один единственный элемент"},"CollectionResponseResourceV2aResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataResourceV2aResponsePublic"}}, "description":"Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"},"DataResourceV2aResponsePublic":{"type":"object","properties":{"range":{"$ref":"#/components/schemas/Range"},"items":{"type":"array","description":"Коллекция элементов","items":{"$ref":"#/components/schemas/ResourceV2aResponsePublic"}}}, "description":"Полезная нагрузка с коллекцией элементов"},"ResourceV2aResponsePublic":{"type":"object","properties":{"id":{"type":"string","description":"ID ресурсной системы","format":"uuid"},"commonName":{"type":"string","description":"Общее наименование (CN) ресурсной системы"},"distinguishedName":{"type":"string","description":"Отличающиеся имя ресурсной системы"},"subjectsCount":{"type":"integer","description":"Количество субъектов ресурсной системы","format":"int64"},"isConnected":{"type":"boolean","description":"Флаг: подключенная ресурсная система"},"isDefault":{"type":"boolean","description":"Флаг: ресурсная система по умолчанию (применяется только к локальным ресурсным системам)"},"created":{"type":"string","description":"Дата и время создания","format":"date-

```

```
time"},"updated":{"type":"string","description":"Дата и время обновления","format":"date-time"}}, {"description":"Ответ: ресурсная система"}, {"ItemResponseResourceV2aResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/ResourceV2aResponsePublic"}}, {"description":"Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, {"CollectionResponseOrganizationalUnitV2aResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataOrganizationalUnitV2aResponsePublic"}}, {"description":"Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"}, {"DataOrganizationalUnitV2aResponsePublic":{"type":"object","properties":{"range":{"$ref":"#/components/schemas/Range"},"items":{"type":"array","description":"Коллекция элементов"},"items":{"$ref":"#/components/schemas/OrganizationalUnitV2aResponsePublic"}}}, {"description":"Полезная нагрузка с коллекцией элементов"}, {"OrganizationalUnitV2aResponsePublic":{"type":"object","properties":{"id":{"type":"string","description":"ID подразделения"},"format":"uuid"},"commonName":{"type":"string","description":"Общее наименование (CN) подразделения"},"distinguishedName":{"type":"string","description":"Отличающееся наименование (DN) подразделения"},"resource":{"$ref":"#/components/schemas/ResourceItemV2aResponsePublic"},"modify":{"type":"string","description":"Дата и время обновления в ресурсной системе"},"format":"date-time"},"updated":{"type":"string","description":"Дата и время обновления","format":"date-time"},"created":{"type":"string","description":"Дата и время создания","format":"date-time"}}, {"description":"Ответ: подразделение"}, {"ItemResponseOrganizationalUnitV2aResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/OrganizationalUnitV2aResponsePublic"}}, {"description":"Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, {"CollectionResponseSubjectResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataSubjectResponsePublic"}}, {"description":"Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"}, {"DataSubjectResponsePublic":{"type":"object","properties":{"range":{"$ref":"#/components/schemas/Range"},"items":{"type":"array","description":"Коллекция элементов"},"items":{"$ref":"#/components/schemas/SubjectResponsePublic"}}}, {"description":"Полезная нагрузка с коллекцией элементов"}, {"CollectionResponseResourceResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataResourceResponsePublic"}}, {"description":"Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"}, {"DataResourceResponsePublic":{"type":"object","properties":{"range":{"$ref":"#/components/schemas/Range"},"items":{"type":"array","description":"Коллекция элементов"},"items":{"$ref":"#/components/schemas/ResourceResponsePublic"}}}, {"description":"Полезная нагрузка с коллекцией элементов"}, {"ResourceResponsePublic":{"type":"object","properties":{"id":{"type":"string","description":"ID ресурсной системы"},"format":"uuid"},"commonName":{"type":"string","description":"Общее наименование (CN) ресурсной системы"},"distinguishedName":{"type":"string","description":"Отличающиеся имя ресурсной системы"},"subjectsCount":{"type":"integer","description":"Количество субъектов ресурсной системы"},"format":"int64"},"isConnected":{"type":"boolean","description":"Флаг: подключенная ресурсная система"},"isDefault":{"type":"boolean","description":"Флаг: ресурсная система по умолчанию (применяется только к локальным ресурсным системам)"},"created":{"type":"string","description":"Дата и время создания","format":"date-time"},"updated":{"type":"string","description":"Дата и время обновления","format":"date-time"}}, {"description":"Ответ: ресурсная система"}, {"ItemResponseResourceResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/ResourceResponsePublic"}}, {"description":"Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, {"CollectionResponseIssuerV2aResponsePublic":{"type":"object","properties":{"status":{"type":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/DataIssuerV2aResponsePublic"}}, {"description":"Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"}, {"DataIssuerV2aResponsePublic":{"type":"object","properties":{"range":{"$ref":"#/components/schemas/Range"},"items":{"type":"array","description":"Коллекция
```

```

элементов", "items": {"$ref": "#/components/schemas/IssuerV2aResponsePublic"}}, "description": "По-
лезная нагрузка с коллекцией
элементов"}, "IssuerV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "fo-
rmat": "uuid"}, "caId": {"type": "string", "format": "uuid"}, "validFrom": {"type": "string", "format": "
date-time"}, "validTo": {"type": "string", "format": "date-
time"}, "name": {"type": "string"}, "title": {"type": "string"}, "subjectKeyIdentifier": {"type": "stri-
ng"}, "authorityKeyIdentifier": {"type": "string"}, "active": {"type": "boolean"}}, "description": "
Коллекция
элементов"}, "AccountV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "d-
escription": "Идентификатор
ресурса", "format": "uuid"}, "login": {"type": "string", "description": "Логин
пользователя"}, "principalName": {"type": "string", "description": "ФИО"}, "role": {"type": "string", "
description": "Роль пользователя", "enum": ["UNKNOWN",
"ADMINISTRATOR", "OPERATOR"]}, "status": {"type": "string", "description": "Статус
пользователя", "enum": ["UNKNOWN",
"ACTIVE", "BLOCKED"]}, "created": {"type": "string", "description": "Дата создания", "format": "date-
time"}, "isLinked": {"type": "boolean", "description": "Флаг: связь с субъектом из ресурсной
системы"}}, "description": "Объект ответа сервиса, содержащий данные об учетной
записи"}, "CollectionResponseAccountV2aResponsePublic": {"type": "object", "properties": {"status":
{"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataAccountV2aRespons-
ePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий коллекцию
элементов"}, "DataAccountV2aResponsePublic": {"type": "object", "properties": {"range": {"$ref": "#/c-
omponents/schemas/Range"}, "items": {"type": "array", "description": " Коллекция
элементов", "items": {"$ref": "#/components/schemas/AccountV2aResponsePublic"}}, "description": "П-
лезная нагрузка с коллекцией
элементов"}, "ItemResponseAccountV2aResponsePublic": {"type": "object", "properties": {"status": {"t-
ype": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/AccountV2aResponsePublic
"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный
элемент"}, "CollectionResponseSyslogV2aResponsePublic": {"type": "object", "properties": {"status":
{"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataSyslogV2aResponse
Public"}}, "description": "Стандартизированный объект ответа сервиса, содержащий коллекцию
элементов"}, "DataSyslogV2aResponsePublic": {"type": "object", "properties": {"range": {"$ref": "#/co-
mponents/schemas/Range"}, "items": {"type": "array", "description": " Коллекция
элементов", "items": {"$ref": "#/components/schemas/SyslogV2aResponsePublic"}}, "description": "По-
лезная нагрузка с коллекцией
элементов"}, "CollectionResponseSyslogResponsePublic": {"type": "object", "properties": {"status": {
"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataSyslogResponsePubl-
ic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий коллекцию
элементов"}, "DataSyslogResponsePublic": {"type": "object", "properties": {"range": {"$ref": "#/compo-
nents/schemas/Range"}, "items": {"type": "array", "description": " Коллекция
элементов", "items": {"$ref": "#/components/schemas/SyslogResponsePublic"}}, "description": "Полез-
ная нагрузка с коллекцией
элементов"}, "CollectionResponseFindConnectionPointV2aResponsePublic": {"type": "object", "propert-
ies": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataFi-
ndConnectionPointV2aResponsePublic"}}, "description": "Стандартизированный объект ответа
сервиса, содержащий коллекцию
элементов"}, "DataFindConnectionPointV2aResponsePublic": {"type": "object", "properties": {"range":
{"$ref": "#/components/schemas/Range"}, "items": {"type": "array", "description": " Коллекция
элементов", "items": {"$ref": "#/components/schemas/FindConnectionPointV2aResponsePublic"}}, "des-
cription": "Полезная нагрузка с коллекцией
элементов"}, "FindConnectionPointV2aResponsePublic": {"type": "object", "properties": {"id": {"type":
"string", "description": "ID точки
подключения", "format": "uuid"}, "title": {"type": "string", "description": "Отображаемое имя точки
подключения"}, "domainType": {"type": "string", "description": "Перечисление: тип
домена", "enum": ["UNKNOWN", "SAMBA_DC", "ALD_PRO", "FREE_IPA", "MS_AD", "RED_ADM",
«ALT_DOMAIN»]}, "connectionAddress": {"type": "string", "description": "Адрес
подключения"}, "useTls": {"type": "boolean", "description": "Флаг: использование
TLS"}, "baseDn": {"type": "string", "description": "Точка подключения
LDAP"}, "username": {"type": "string", "description": "Имя доменного
пользователя"}, "status": {"type": "string", "description": "Перечисление: статусы синхронизации
ресурсной системы", "enum": ["UNKNOWN",

```

```

"NONE", "IN_QUEUE", "IN_PROGRESS", "SUCCESS", "ERROR"]}, "resourceId": {"type": "string", "description": "ID ресурсной системы", "format": "uuid"}, "created": {"type": "string", "description": "Дата и время создания", "format": "date-time"}, "updated": {"type": "string", "description": "Дата и время обновления", "format": "date-time"}}, "description": "Ответ: точка подключения"}, "ItemResponseFindConnectionPointV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/FindConnectionPointV2aResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, "CollectionResponseFindConnectionPointResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataFindConnectionPointResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"}, "DataFindConnectionPointResponsePublic": {"type": "object", "properties": {"range": {"$ref": "#/components/schemas/Range"}, "items": {"type": "array", "description": "Коллекция элементов", "items": {"$ref": "#/components/schemas/FindConnectionPointResponsePublic"}}}, "description": "Полезная нагрузка с коллекцией элементов"}, "FindConnectionPointResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "ID точки подключения", "format": "uuid"}, "title": {"type": "string", "description": "Отображаемое имя точки подключения"}, "domainType": {"type": "string", "description": "Перечисление: тип домена", "enum": ["UNKNOWN", "SAMBA_DC", "ALD_PRO", "FREE_IPA", "MS_AD", "RED_ADM"]}, "connectionAddress": {"type": "string", "description": "Адрес подключения"}, "useTls": {"type": "boolean", "description": "Флаг: использование TLS"}, "baseDn": {"type": "string", "description": "Точка подключения LDAP"}, "username": {"type": "string", "description": "Имя доменного пользователя"}, "status": {"type": "string", "description": "Перечисление: статусы синхронизации ресурсной системы", "enum": ["UNKNOWN", "NONE", "IN_QUEUE", "IN_PROGRESS", "SUCCESS", "ERROR"]}, "resourceId": {"type": "string", "description": "ID ресурсной системы", "format": "uuid"}, "created": {"type": "string", "description": "Дата и время создания", "format": "date-time"}, "updated": {"type": "string", "description": "Дата и время обновления", "format": "date-time"}}, "description": "Ответ: точка подключения"}, "ItemResponseFindConnectionPointResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/FindConnectionPointResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, "GetApiVersionResponse": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/VersionResponse"}}, "VersionResponse": {"type": "object", "properties": {"version": {"type": "string", "description": "Версия модуля"}}, "description": "Объект ответа, содержащий версию модуля"}, "CollectionResponseFindAllCertificateV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataFindAllCertificateV2aResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"}, "DataFindAllCertificateV2aResponsePublic": {"type": "object", "properties": {"range": {"$ref": "#/components/schemas/Range"}, "items": {"type": "array", "description": "Коллекция элементов", "items": {"$ref": "#/components/schemas/FindAllCertificateV2aResponsePublic"}}}, "description": "Полезная нагрузка с коллекцией элементов"}, "FindAllCertificateV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Уникальный идентификатор сертификата", "format": "uuid"}, "issuerId": {"type": "string", "description": "Уникальный идентификатор издающего сертификата", "format": "uuid"}, "issuerFingerprint": {"type": "string", "description": "Цифровой отпечаток издающего сертификат"}, "serialnumber": {"type": "string", "description": "Серийный номер сертификата"}, "fingerprint": {"type": "string", "description": "Цифровой отпечаток сертификата"}, "issuerDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Отличительное имя владельца", "items": {"type": "string", "description": "Отличительное имя владельца"}}, "description": "Отличительное имя владельца"}, "subjectDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Отличительное имя субъекта", "items": {"type": "string", "description": "Отличительное имя субъекта"}}, "description": "Отличительное имя субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "descripti

```

```

on": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя субъекта"}}, "description": "Дополнительное имя субъекта", "name": {"type": "string", "description": "Имя сертификата (формируется из CN, разделитель \"_\")"}, "templateId": {"type": "string", "description": "Идентификатор шаблона, по которому выпущен сертификат", "format": "uuid"}, "templateName": {"type": "string", "description": "Название шаблона, по которому выпущен сертификат"}, "validFrom": {"type": "string", "description": "Дата начала действия сертификата", "format": "date-time"}, "validTo": {"type": "string", "description": "Дата окончания действия сертификата", "format": "date-time"}, "status": {"type": "string", "description": "Перечисление: статус сертификата", "enum": ["UNKNOWN", "ACTIVE", "HOLD", "REVOKE", "REQUEST"]}, "isValid": {"type": "boolean", "description": "Сертификат действителен"}, "isExpired": {"type": "boolean", "description": "Сертификат истек"}, "actions": {"$ref": "#/components/schemas/CertificateDownloadActionsV2aPublic"}, "revocation": {"$ref": "#/components/schemas/RevocationInfoV2aPublic"}, "hashAlgorithm": {"type": "string", "description": "Перечисление Hash алгоритмов", "enum": ["UNKNOWN", "SHA1", "SHA256", "SHA384", "SHA512", "SHA3_256", "SHA3_384", "SHA3_512", "RSASSA_PSS", "MD5", "GOST_R_34_11_2012", "UNKNOWN"]}, "keyAlgorithm": {"$ref": "#/components/schemas/EKeyAlgorithmV2aPublic"}, "keyBits": {"type": "integer", "description": "Размерность ключа", "format": "int32"}, "subjectKeyIdentifier": {"type": "string", "description": "Идентификатор ключа субъекта"}, "authorityKeyIdentifier": {"type": "string", "description": "Идентификатор ключа УЦ"}, "created": {"type": "string", "description": "Дата создания шаблона", "format": "date-time"}, "updated": {"type": "string", "description": "Дата обновления шаблона", "format": "date-time"}}, "description": "Объект ответа сервиса, содержащий данные сертификата", "CollectionResponseFindCertificateAuthorityV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataFindCertificateAuthorityV2aResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"}, "CryptographyProviderDataV2aPublic": {"type": "object", "properties": {"cryptographyProvider": {"type": "string", "description": "Перечисление: криптопровайдеры", "enum": ["UNKNOWN", "DEFAULT", "CRYPTO_PRO"]}, "isAvailable": {"type": "boolean", "description": "Доступность криптопровайдера"}}, "description": "POJO: Криптопровайдер"}, "DataFindCertificateAuthorityV2aResponsePublic": {"type": "object", "properties": {"range": {"$ref": "#/components/schemas/Range"}, "items": {"type": "array", "description": "Коллекция элементов", "items": {"$ref": "#/components/schemas/FindCertificateAuthorityV2aResponsePublic"}}}, "description": "Полезная нагрузка с коллекцией элементов"}, "FindCertificateAuthorityV2aResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Уникальный идентификатор центра сертификации", "format": "uuid"}, "isActive": {"type": "boolean", "description": "Флаг: активен"}, "active": {"type": "boolean", "description": "Флаг: активен", "deprecated": true}, "isManagement": {"type": "boolean", "description": "Флаг: Технологический ЦС"}, "management": {"type": "boolean", "description": "Флаг: Технологический ЦС", "deprecated": true}, "certificate": {"$ref": "#/components/schemas/FindAllCertificateV2aResponsePublic"}, "chain": {"$ref": "#/components/schemas/CertificateChainV2aPublic"}, "certificateCount": {"type": "integer", "description": "Количество сертификатов, выпущенных центром сертификации", "format": "int64"}, "title": {"type": "string", "description": "Отображаемое имя"}, "cryptographyProviders": {"type": "object", "additionalProperties": {"$ref": "#/components/schemas/CryptographyProviderDataV2aPublic"}, "description": "Конфигурация криптопровайдеров для ЦС"}, "primaryCryptographyProvider": {"$ref": "#/components/schemas/CryptographyProviderDataV2aPublic"}, "isAvailable": {"type": "boolean", "description": "Доступность ЦС"}, "created": {"type": "string", "description": "Дата создания сертификата", "format": "date-time"}, "updated": {"type": "string", "description": "Дата обновления сертификата", "format": "date-time"}}, "description": "Объект ответа сервиса, содержащий данные центра сертификации"}, "ItemResponseFindCertificateAuthorityV2aResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/FindCertificateAuthorityV2aResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, "CollectionResponseFindAllCertificateResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataFindAllCertificateResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса,

```

содержащий коллекцию элементов"}, "DataFindAllCertificateResponsePublic": {"type": "object", "properties": {"range": {"\$ref": "#/components/schemas/Range"}, "items": {"type": "array", "description": "Коллекция элементов", "items": {"\$ref": "#/components/schemas/FindAllCertificateResponsePublic"}}}, "description": "Полезная нагрузка с коллекцией элементов"}, "FindAllCertificateResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Уникальный идентификатор сертификата", "format": "uuid"}, "issuerId": {"type": "string", "description": "Уникальный идентификатор издающего сертификата", "format": "uuid"}, "issuerFingerprint": {"type": "string", "description": "Цифровой отпечаток издающего сертификат"}, "serialnumber": {"type": "string", "description": "Серийный номер сертификата"}, "fingerprint": {"type": "string", "description": "Цифровой отпечаток сертификата"}, "issuerDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Отличительное имя владельца", "items": {"type": "string", "description": "Отличительное имя владельца"}}, "description": "Отличительное имя владельца"}, "subjectDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Отличительное имя субъекта", "items": {"type": "string", "description": "Отличительное имя субъекта"}}, "description": "Отличительное имя субъекта"}, "subjectAltName": {"type": "object", "additionalProperties": {"type": "array", "description": "Дополнительное имя субъекта", "items": {"type": "string", "description": "Дополнительное имя субъекта"}}, "description": "Дополнительное имя субъекта"}, "name": {"type": "string", "description": "Имя сертификата (формируется из CN, разделитель \"_\")"}, "templateId": {"type": "string", "description": "Идентификатор шаблона, по которому выпущен сертификат", "format": "uuid"}, "templateName": {"type": "string", "description": "Название шаблона, по которому выпущен сертификат"}, "validFrom": {"type": "string", "description": "Дата начала действия сертификата", "format": "date-time"}, "validTo": {"type": "string", "description": "Дата окончания действия сертификата", "format": "date-time"}, "status": {"type": "string", "description": "Перечисление: статус сертификата", "enum": ["UNKNOWN", "ACTIVE", "HOLD", "REVOKE", "REQUEST"]}, "isValid": {"type": "boolean", "description": "Сертификат действителен"}, "isExpired": {"type": "boolean", "description": "Сертификат истек"}, "actions": {"\$ref": "#/components/schemas/CertificateDownloadActionsPublic"}, "revocation": {"\$ref": "#/components/schemas/RevocationInfoPublic"}, "hashAlgorithm": {"type": "string", "description": "Перечисление Hash алгоритмов", "enum": ["UNKNOWN", "SHA1", "SHA256", "SHA384", "SHA512", "SHA3_256", "SHA3_384", "SHA3_512", "RSASSA_PSS", "MD5", "GOST_R_34_11_2012", "UNKNOWN"]}, "keyAlgorithm": {"\$ref": "#/components/schemas/EKeyAlgorithmPublic"}, "keyBits": {"type": "integer", "description": "Размерность ключа", "format": "int32"}, "subjectKeyIdentifier": {"type": "string", "description": "Идентификатор ключа субъекта"}, "authorityKeyIdentifier": {"type": "string", "description": "Идентификатор ключа УЦ"}, "created": {"type": "string", "description": "Дата создания шаблона", "format": "date-time"}, "updated": {"type": "string", "description": "Дата обновления шаблона", "format": "date-time"}}, "description": "Объект ответа сервиса, содержащий данные сертификата"}, "RevocationInfoPublic": {"type": "object", "properties": {"date": {"type": "string", "description": "Дата отзыва", "format": "date-time"}, "number": {"type": "integer", "description": "Номер причины отзыва", "format": "int32"}, "value": {"type": "string", "description": "Значение причины отзыва"}}, "description": "ПОЖО: Данные об отзыве"}, "CertificateChainPublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Уникальный идентификатор сертификата", "format": "uuid"}, "name": {"type": "string", "description": "Имя сертификата (формируется из CN, разделитель \"_\")"}, "subjectDN": {"type": "object", "additionalProperties": {"type": "array", "description": "Subject DN", "items": {"type": "string", "description": "Subject DN"}}, "description": "Subject DN"}, "issuer": {"\$ref": "#/components/schemas/CertificateChainPublic"}}, "description": "ПОЖО: цепочка сертификатов"}, "CertificateResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Уникальный идентификатор сертификата", "format": "uuid"}, "chain": {"\$ref": "#/components/schemas/CertificateChainPublic"}, "serialnumber": {"type": "string", "description": "Серийный номер сертификата"}, "fingerprint": {"type": "string", "description": "Фингерпринт

```

сертификата"},"name":{"type":"string","description":"Имя сертификата (формируется из CN,
разделитель \"_\")"},"issuerId":{"type":"string","description":"Уникальный идентификатор
издающего
сертификата","format":"uuid"},"issuerDN":{"type":"object","additionalProperties":{"type":"arra
y","description":"Issuer DN","items":{"type":"string","description":"Issuer
DN"}}},"description":"Issuer
DN"},"subjectDN":{"type":"object","additionalProperties":{"type":"array","description":"Subjec
t DN","items":{"type":"string","description":"Subject DN"}}},"description":"Subject
DN"},"subjectAltName":{"type":"object","additionalProperties":{"type":"array","description":"S
ubject AltName","items":{"type":"string","description":"Subject
AltName"}}},"description":"Subject
AltName"},"keyAlgorithm":{"$ref":"#/components/schemas/EKeyAlgorithmPublic"},"hashAlgorithm":{"
type":"string","description":"Перечисление Hash алгоритмов","enum":["UNKNOWN",
"SHA1","SHA256","SHA384","SHA512","SHA3_256","SHA3_384","SHA3_512","RSASSA_PSS","MD5","GOST_R_
34_11_2012","UNKNOWN"]},"keyBits":{"type":"integer","description":"Размерность
ключа","format":"int32"},"templateId":{"type":"string","description":"Идентификатор шаблона,
по которому выпущен
сертификат","format":"uuid"},"templateName":{"type":"string","description":"Название шаблона,
по которому выпущен сертификат"},"type":{"type":"string","description":"Перечисление типов
сертификата","enum":["UNKNOWN",
"CERTIFICATE","ROOT_CA","SUB_CA"]},"validFrom":{"type":"string","description":"Дата начала
действия сертификата","format":"date-time"},"validTo":{"type":"string","description":"Дата
окончания действия сертификата","format":"date-
time"},"status":{"type":"string","description":"Перечисление: статус
сертификата","enum":["UNKNOWN",
"ACTIVE","HOLD","REVOKE","REQUEST"]},"isValid":{"type":"boolean","description":"Сертификат
действителен"},"isExpired":{"type":"boolean","description":"Сертификат
истек"},"actions":{"$ref":"#/components/schemas/CertificateDownloadActionsPublic"},"publicKey"
":{"type":"string","description":"Публичный ключ
сертификата"},"certificateType":{"type":"string","description":"Тип
сертификата"},"version":{"type":"integer","description":"Версия
сертификата","format":"int32"},"subjectKeyIdentifier":{"type":"string","description":"Идентифи
катор ключа
сертификата"},"authorityKeyIdentifier":{"type":"string","description":"Идентификатор ключа
издателя"},"keyUsages":{"type":"array","description":"Разрешенные использования
ключа","items":{"$ref":"#/components/schemas/KeyUsageDictResponsePublic"},"extendedKeyUsages"
":{"type":"array","description":"Разрешенные расширенные использования
ключа","items":{"$ref":"#/components/schemas/ExtendedKeyUsageResponsePublic"},"ca":{"type":"b
oolean","description":"Флаг: сертификат
ЦС"},"revocation":{"$ref":"#/components/schemas/RevocationInfoPublic"},"aiaUrls":{"type":"arra
y","description":"Точки распространения AIA","items":{"type":"string","description":"Точки
распространения AIA"},"ocspUrls":{"type":"array","description":"URL сервисов
OCSP","items":{"type":"string","description":"URL сервисов
OCSP"},"crlUrls":{"type":"array","description":"Точки распространения
CRL","items":{"type":"string","description":"Точки распространения
CRL"},"deltaCrlUrls":{"type":"array","description":"Точки распространения Delta
CRL","items":{"type":"string","description":"Точки распространения Delta
CRL"},"userId":{"type":"string","description":"Идентификатор
пользователя","format":"uuid"},"subjectId":{"type":"string","description":"Идентификатор
субъекта","format":"uuid"},"created":{"type":"string","description":"Дата создания
шаблона","format":"date-time"},"updated":{"type":"string","description":"Дата обновления
шаблона","format":"date-time"},"description":"Ответ сервиса: данные
сертификата"},"ExtendedKeyUsageResponsePublic":{"type":"object","properties":{"id":{"type":"st
ring","description":"Идентификатор расширенного использования
ключа","format":"uuid"},"oid":{"type":"string","description":"Идентификатор OID расширенного
использования ключа"},"value":{"type":"string","description":"Наименование расширенного
использования ключа"},"description":{"type":"string","description":"Описание использования
ключа"},"description":"Разрешенные расширенные использования
ключа"},"ItemResponseCertificateResponsePublic":{"type":"object","properties":{"status":{"type
":"integer","format":"int32"},"data":{"$ref":"#/components/schemas/CertificateResponsePublic"}
},"description":"Стандартизированный объект ответа сервиса, содержащий один единственный

```

```

элемент"}, "KeyUsageDictResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "format": "uuid"}, "code": {"type": "string", "description": "Перечисление использования ключа", "enum": ["UNKNOWN", "DIGITAL_SIGNATURE", "NON_REPUDIATION", "KEY_ENCIPHERMENT", "DATA_ENCIPHERMENT", "KEY_AGREEMENT", "KEY_CERT_SIGN", "CRL_SIGN", "ENCIPHER_ONLY", "DECIPHER_ONLY"]}, "description": {"type": "string", "value": {"type": "string"}}, "description": "Разрешенные использования ключа"}, "CollectionResponseFindCertificateAuthorityResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/DataFindCertificateAuthorityResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий коллекцию элементов"}, "CryptographyProviderDataPublic": {"type": "object", "properties": {"cryptographyProvider": {"type": "string", "description": "Перечисление: криптопровайдеры", "enum": ["UNKNOWN", "DEFAULT", "CRYPTO_PRO"]}, "isAvailable": {"type": "boolean", "description": "Доступность криптопровайдера"}}, "description": "POJO: Криптопровайдер"}, "DataFindCertificateAuthorityResponsePublic": {"type": "object", "properties": {"range": {"$ref": "#/components/schemas/Range"}, "items": {"type": "array", "description": "Коллекция элементов", "items": {"$ref": "#/components/schemas/FindCertificateAuthorityResponsePublic"}}, "description": "Полезная нагрузка с коллекцией элементов"}, "FindCertificateAuthorityResponsePublic": {"type": "object", "properties": {"id": {"type": "string", "description": "Уникальный идентификатор центра сертификации", "format": "uuid"}, "isActive": {"type": "boolean", "description": "Флаг: активен"}, "active": {"type": "boolean", "description": "Флаг: активен", "deprecated": true}, "isManagement": {"type": "boolean", "description": "Флаг: Технологический ЦС"}, "management": {"type": "boolean", "description": "Флаг: Технологический ЦС", "deprecated": true}, "certificate": {"$ref": "#/components/schemas/FindAllCertificateResponsePublic"}, "chain": {"$ref": "#/components/schemas/CertificateChainPublic"}, "certificateCount": {"type": "integer", "description": "Количество сертификатов, выпущенных центром сертификации", "format": "int64"}, "title": {"type": "string", "description": "Отображаемое имя"}, "cryptographyProviders": {"type": "object", "additionalProperties": {"$ref": "#/components/schemas/CryptographyProviderDataPublic"}, "description": "Конфигурация криптопровайдеров для ЦС"}, "primaryCryptographyProvider": {"$ref": "#/components/schemas/CryptographyProviderDataPublic"}, "isAvailable": {"type": "boolean", "description": "Доступность ЦС"}, "created": {"type": "string", "description": "Дата создания сертификата", "format": "date-time"}, "updated": {"type": "string", "description": "Дата обновления сертификата", "format": "date-time"}}, "description": "Объект ответа сервиса, содержащий данные центра сертификации"}, "ItemResponseFindCertificateAuthorityResponsePublic": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "data": {"$ref": "#/components/schemas/FindCertificateAuthorityResponsePublic"}}, "description": "Стандартизированный объект ответа сервиса, содержащий один единственный элемент"}, "securitySchemes": {"X-API-KEY": {"type": "apiKey", "description": "API Key", "name": "X-API-KEY", "in": "header"}, "token": {"type": "apiKey", "description": "Token", "name": "token", "in": "header"}, "X-AECA-User-ID": {"type": "apiKey", "description": "ID", "name": "X-AECA-User-ID", "in": "header"}, "X-AECA-User-Login": {"type": "apiKey", "description": "Login", "name": "X-AECA-User-Login", "in": "header"}, "X-AECA-User-Principal": {"type": "apiKey", "description": "Principal name", "name": "X-AECA-User-Principal", "in": "header"}, "X-AECA-User-Role": {"type": "apiKey", "description": "Role", "name": "X-AECA-User-Role", "in": "header"}, "X-AECA-User-Authorities": {"type": "apiKey", "description": "Authorities", "name": "X-AECA-User-Authorities", "in": "header"}, "X-AECA-User-Organizational-Unit-Rights": {"type": "apiKey", "description": "Organizational unit rights", "name": "X-AECA-User-Organizational-Unit-Rights", "in": "header"}, "X-AECA-User-Security-Group-Rights": {"type": "apiKey", "description": "Security group rights", "name": "X-AECA-User-Security-Group-Rights", "in": "header"}, "X-AECA-User-Resource-Rights": {"type": "apiKey", "description": "Resource rights", "name": "X-AECA-User-Resource-Rights", "in": "header"}, "X-User-Session-Id": {"type": "apiKey", "description": "Session ID", "name": "X-User-Session-Id", "in": "header"}, "X-AECA-Subject-Id": {"type": "apiKey", "description": "Subject ID", "name": "X-AECA-Subject-Id", "in": "header"}, "X-AECA-Organizational-unit-Id": {"type": "apiKey", "description": "Organizational unit id", "name": "X-AECA-Organizational-unit-Id", "in": "header"}, "X-AECA-Security-group-Ids": {"type": "apiKey", "description": "Security group ids", "name": "X-AECA-Security-group-Ids", "in": "header"}}}

```

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

ОС	–	Операционная система
ПО	–	Программное обеспечение
ЦС	–	Центр сертификации
API	–	Application Programming Interface
CRL	–	Certificate Revocation List
AIA	–	Authority Information Access
URL	–	Uniform Resource Locator

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

[illegible]